

TABLE OF CONTENTS

1			1
2	\$ASSEMBLY_AND_OPERATION_INFORMATION.AGC	# PP. 1-27	2
3	\$TAGS_FOR_RELATIVE_SETLOC.AGC	# PP. 28-37	3
4	\$CONTROLLED_CONSTANTS.AGC	# PP. 38-53	4
5	\$INPUT_OUTPUT_CHANNEL_BIT_DESCRIPTIONS.AGC	# PP. 54-60	5
6	\$FLAGWORD_ASSIGNMENTS.AGC	# PP. 61-88	6
7	\$ERASABLE_ASSIGNMENTS.AGC	# PP. 90-152	7
8	\$INTERRUPT_LEAD_INS.AGC	# PP. 153-154	8
9	\$T4RUPT_PROGRAM.AGC	# PP. 155-189	9
10	\$RCS_FAILURE_MONITOR.AGC	# PP. 190-192	10
11	\$DOWNLINK_LISTS.AGC	# PP. 193-205	11
12	\$AGS_INITIALIZATION.AGC	# PP. 206-210	12
13	\$FRESH_START_AND_RESTART.AGC	# PP. 211-237	13
14	\$RESTART_TABLES.AGC	# PP. 238-243	14
15	\$AOTMARK.AGC	# PP. 244-261	15
16	\$EXTENDED_VERBS.AGC	# PP. 262-300	16
17	\$PINBALL_NOUN_TABLES.AGC	# PP. 301-319	17
18	\$LEM_GEOMETRY.AGC	# PP. 320-325	18
19	\$IMU_COMPENSATION_PACKAGE.AGC	# PP. 326-337	19
20	\$R63.AGC	# PP. 338-341	20
21	\$ATTITUDE_MANEUVER_ROUTINE.AGC	# PP. 342-363	21
22	\$GIMBAL_LOCK_AVOIDANCE.AGC	# P. 364	22
23	\$KALCMANU_STEERING.AGC	# PP. 365-369	23
24	\$SYSTEM_TEST_STANDARD_LEAD_INS.AGC	# PP. 370-372	24
25	\$IMU_PERFORMANCE_TEST_2.AGC	# PP. 373-381	25
26	\$IMU_PERFORMANCE_TESTS_4.AGC	# PP. 382-389	26
27	\$PINBALL_GAME_BUTTONS_AND_LIGHTS.AGC	# PP. 390-471	27
28	\$R60_62.AGC	# PP. 472-485	28
29	\$S-BAND_ANTENNA_FOR_LM.AGC	# PP. 486-489	29
30	\$RADAR_LEADIN_ROUTINES.AGC	# PP. 490-491	30
31	\$P20-P25.AGC	# PP. 492-614	31
32	\$P30_P37.AGC	# PP. 615-617	32
33	\$P32-P35_P72-P75.AGC	# PP. 618-650	33
34	\$LAMBERT_AIMPOINT_GUIDANCE.AGC	# PP. 651-653	34
35	\$GROUND_TRACKING_DETERMINATION_PROGRAM.AGC	# PP. 654-657	35
36	\$P34-35_P74-75.AGC	# PP. 658-702	36
37	\$R31.AGC	# PP. 703-708	37
38	\$P76.AGC	# PP. 709-711	38
39	\$R30.AGC	# PP. 712-722	39
40	\$STABLE_ORBIT.AGC	# PP. 723-730	40
41	\$BURN_BABY_BURN--MASTER_IGNITION_ROUTINE.AGC	# PP. 731-751	41
42	\$P40-P47.AGC	# PP. 752-784	42
43	\$THE_LUNAR_LANDING.AGC	# PP. 785-792	43
44	\$THROTTLE_CONTROL_ROUTINES.AGC	# PP. 793-797	44
45	\$LUNAR_LANDING_GUIDANCE_EQUATIONS.AGC	# PP. 798-828	45
46	\$P70-P71.AGC	# PP. 829-837	46
47	\$P12.AGC	# PP. 838-842	47
48	\$ASCENT_GUIDANCE.AGC	# PP. 843-856	48
49	\$SERVICER.AGC	# PP. 857-897	49
50	\$LANDING_ANALOG_DISPLAYS.AGC	# PP. 898-907	50
51	\$FINDCDUW--GUIDAP_INTERFACE.AGC	# PP. 908-925	51
52	\$P51-P53.AGC	# PP. 926-983	52
53	\$LUNAR_AND_SOLAR_EPHEMERIDES_SUBROUTINES.AGC	# PP. 984-987	53
54	\$DOWN_TELEMETRY_PROGRAM.AGC	# PP. 988-997	54
55	\$INTER-BANK_COMMUNICATION.AGC	# PP. 998-1001	55
56	\$INTERPRETER.AGC	# PP. 1002-1094	56
57	\$FIXED_FIXED_CONSTANT_POOL.AGC	# PP. 1095-1099	57
58	\$INTERPRETIVE_CONSTANT.AGC	# PP. 1100-1101	58
59	\$SINGLE_PRECISION_SUBROUTINES.AGC	# P. 1102	59
60	\$EXECUTIVE.AGC	# PP. 1103-1116	60



1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

\$WAITLIST.AGC # PP. 1117-1132
\$LATITUDE_LONGITUDE_SUBROUTINES.AGC # PP. 1133-1139
\$PLANETARY_INERTIAL_ORIENTATION.AGC # PP. 1140-1148
\$MEASUREMENT_INCORPORATION.AGC # PP. 1149-1158
\$CONIC_SUBROUTINES.AGC # PP. 1159-1204
\$INTEGRATION_INITIALIZATION.AGC # PP. 1205-1226
\$ORBITAL_INTEGRATION.AGC # PP. 1227-1248
\$INFLIGHT_ALIGNMENT_ROUTINES.AGC # PP. 1249-1258
\$POWERED_FLIGHT_SUBROUTINES.AGC # PP. 1259-1267
\$TIME_OF_FREE_FALL.AGC # PP. 1268-1283
\$AGC_BLOCK_TWO_SELF_CHECK.AGC # PP. 1284-1293
\$PHASE_TABLE_MAINTENANCE.AGC # PP. 1294-1302
\$RESTARTS_ROUTINE.AGC # PP. 1303-1308
\$IMU_MODE_SWITCHING_ROUTINES.AGC # PP. 1309-1337
\$KEYRUPT_UPRUPT.AGC # PP. 1338-1340
\$DISPLAY_INTERFACE_ROUTINES.AGC # PP. 1341-1373
\$SERVICE_ROUTINES.AGC # PP. 1374-1380
\$ALARM_AND_ABORT.AGC # PP. 1381-1385
\$UPDATE_PROGRAM.AGC # PP. 1386-1396
\$RTB_OP_CODES.AGC # PP. 1397-1402
\$T6-RUPT_PROGRAMS.AGC # PP. 1403-1405
\$DAP_INTERFACE_SUBROUTINES.AGC # PP. 1406-1409
\$DAPIDLER_PROGRAM.AGC # PP. 1410-1420
\$P-AXIS_RCS_AUTOPILOT.AGC # PP. 1421-1441
\$Q_R-AXIS_RCS_AUTOPILOT.AGC # PP. 1442-1459
\$TJET_LAW.AGC # PP. 1460-1469
\$KALMAN_FILTER.AGC # PP. 1470-1471
\$TRIM_GIMBAL_CNTRL_SYSTEM.AGC # PP. 1472-1484
\$AOSTASK_AND_AOSJOB.AGC # PP. 1485-1506
\$SPS_BACK-UP_RCS_CONTROL.AGC # PP. 1507-1510



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

THIS LGC PROGRAM IS INTENDED FOR USE IN THE LM DURING THE MANNED LUNAR LANDING MISSION OR ANY SUBSET THEREOF.
THE DETAILS OF IMPLEMENTATION ARE SPECIFIED IN REPORT R-567, AS AMENDED.
#
GUIDANCE SYSTEM OPERATIONS PLAN
#
FOR MANNED LM EARTH ORBITAL AND LUNAR MISSIONS
#
USING PROGRAM LUMINARY
THIS PROGRAM AND R-567 HAVE BEEN PREPARED BY THE INSTRUMENTATION LABORATORY, MASSACHUSETTS INSTITUTE OF
TECHNOLOGY, 75 CAMBRIDGE PARKWAY, CAMBRIDGE, MASSACHUSETTS, UNDER PROJECT 55-238-70, SPONSORED BY THE MANNED
SPACECRAFT CENTER OF THE NATIONAL AERONAUTICS AND SPACE ADMINISTRATION, CONTRACT NAS 9-4065.
#
THIS PROGRAM IS REFERRED TO AS LUMINARY 1A

TABLE OF LOG CARDS

ASSEMBLY AND OPERATION INFORMATION
TAGS FOR RELATIVE SETLOC AND BLANK BANK CARDS
CONTROLLED CONSTANTS
INPUT/OUTPUT CHANNEL BIT DESCRIPTIONS
FLAGWORD ASSIGNMENTS
SUBROUTINE CALLS

TABLE OF SUBROUTINE LOG SECTIONS
LUMERASE

ERASABLE ASSIGNMENTS
LNYAIDE

INTERRUPT LEAD INS

T4RUPT PROGRAM
RCS FAILURE MONITOR
DOWNLINK LISTS

AGS INITIALIZATION
FRESH START AND RESTART
RESTART TABLES

AOTMARK
EXTENDED VERBS
PINBALL NOUN TABLES

LEM GEOMETRY
IMU COMPENSATION PACKAGE
R63

ATTITUDE MANEUVER ROUTINE
GIMBAL LOCK AVOIDANCE
KALCMANU STEERING

SYSTEM TEST STANDARD LEAD INS
IMU PERFORMANCE TESTS 2
IMU PERFORMANCE TESTS 4

PINBALL GAMES BUTTONS AND LIGHTS
R60,R62
S-BAND ANTENNA FOR LM

LEMP20S
RADAR LEADIN ROUTINES
P20-P25

LEMP30S
P30,P37
P32-P35, P72-P75

GENERAL LAMBERT AIMPOINT GUIDANCE
KISSING

GROUND TRACKING DETERMINATION PROGRAM - P21
P34-P35, P74-P75
R31
P76

R30
STABLE ORBIT - P38-P39

1



1				1
2	#	KALMAN FILTER		2
3	#	TRIM GIMBAL CONTROL SYSTEM		3
4	#	AOSTASK AND AOSJOB		4
5	#	SPS BACK-UP RCS CONTROL		5
6	#			6
7	#	SYMBOL TABLE LISTING		7
8	#	UNREFERENCED SYMBOL LISTING		8
9	#	ERASABLE & EQUALS CROSS-REFERENCE TABLE		9
10	#	SUMMARY OF SYMBOL TABLE LISTINGS		10
11	#	MEMORY TYPE & AVAILABILITY DISPLAY		11
12	#	COUNT TABLE		12
13	#	PARAGRAPHS GENERATED FOR THIS ASSEMBLY		13
14	#	OCTAL LISTING		14
15	#	OCCUPIED LOCATIONS TABLE		15
16	#	SUBROS CALLED & PROGRAM STATUS		16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

VERB LIST FOR LUMINARY

REGULAR VERBS

00 NOT IN USE

01 DISPLAY OCTAL COMP 1 IN R1

02 DISPLAY OCTAL COMP 2 IN R1

03 DISPLAY OCTAL COMP 3 IN R1

04 DISPLAY OCTAL COMP 1,2 IN R1,R2

05 DISPLAY OCTAL COMP 1,2,3 IN R1,R2,R3

06 DISPLAY DECIMAL IN R1 OR R1,R2 OR R1,R2,R3

07 DISPLAY DP DECIMAL IN R1,R2 (TEST ONLY)

08

09

10

11 MONITOR OCTAL COMP 1 IN R1

12 MONITOR OCTAL COMP 2 IN R1

13 MONITOR OCTAL COMP 3 IN R1

14 MONITOR OCTAL COMP 1,2 IN R1,R2

15 MONITOR OCTAL COMP 1,2,3 IN R1,R2,R3

16 MONITOR DECIMAL IN R1 OR R1,R2 OR R1,R2,R3

17 MONITOR DP DECIMAL IN R1,R2 (TEST ONLY)

18

19

20

21 LOAD COMPONENT 1 INTO R1

22 LOAD COMPONENT 2 INTO R2

23 LOAD COMPONENT 3 INTO R3

24 LOAD COMPONENT 1,2, INTO R1,R2

25 LOAD COMPONENT 1,2,3 INTO R1,R2,R3

26

27 DISPLAY FIXED MEMORY

28

29

30 REQUEST EXECUTIVE

31 REQUEST WAITLIST

32 RECYCLE PROGRAM

33 PROCEED WITHOUT DSKY INPUTS

34 TERMINATE FUNCTION

35 TEST LIGHTS

36 REQUEST FRESH START

37 CHANGE PROGRAM (MAJOR MODE)

38

39

EXTENDED VERBS

40 ZERO CDU-S
41 COARSE ALIGN CDU-S
42 FINE ALIGN IMU
43 LOAD IMU ATT ERROR METERS
44 TERMINATE RR CONTINUOUS DESIGNATE (V41N72 OPTION 2)
45
46
47 INITIALIZE AGS (R47)
48 REQUEST DAP DATA LOAD ROUTINE (R03)
49 REQUEST CREW DEFINED MANEUVER ROUTINE (R62)
50 PLEASE PERFORM
51
52 MARK X-RETICLE
53 MARK Y-RETICLE
54 MARK X OR Y-RETICLE
55 INCREMENT AGC TIME (DECIMAL)
56 TERMINATE TRACKING (P20 + P25)
57 PERMIT LANDING RADAR UPDATES
58 INHIBIT LANDING RADAR UPDATES
59
60 COMMAND LR TO POSITON 2.
61 DISPLAY DAP FOLLOWING ATTITUDE ERRORS.
62 DISPLAY TOTAL ATTITUDE ERRORS WITH RESPECT TO NOUN 22.
63 SAMPLE RADAR ONCE PER SECOND (R04).
64 REQUEST S-BAND ANTENNA ROUTINE (R05).
65 DISABLE U AND V JET FIRINGS DURING DPS BURNS.
66 VEHICLES ARE ATTACHED. MOVE THIS VEHICLE STATE TO OTHER VEHICLE.
67 DISPLAY W MATRIX
68
69 CAUSE RESTART
70 UPDATE LIFTOFF TIME
71 UNIVERSAL UPDATE-BLOCK ADR
72 UNIVERSAL UPDATE-SINGLE ADR
73 UPDATE AGC TIME (OCTAL)
74 INITIALIZE ERASABLE DUMP VIA DOWNLINK
75 ENABLE U AND V JET FIRINGS DURING DPS BURNS.
76 MINIMUM IMPUSE COMMAND MODE.
77 RATE COMMAND AND ATTITUDE HOLD MODE
78 LR SPURIOUS RETURN TEST START (R77)
79 LR SPURIOUS RETURN TEST STOP
80 UPDATE LEM STATE VECTOR
81 UPDATE CSM STATE VECTOR
82 REQUEST ORBIT PARAM DISPLAY (R30)
83 REQUEST REND PARAM DISPLAY (R31)
84
85 DISPLAY RR LOS AZ AND ELEV
86
87



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

ASSEMBLY_AND_OPERATION_INFORMATION

88
89 REQUEST RENDEZVOUS FINAL ATTITUDE ROUTINE (R63)
90 REQUEST RENDEZVOUS OUT OF PLANE DISPLAY ROUTINE (R36)
91 DISPLAY BANK SUM
92 OPERATE IMU PERFORMANCE TEST (P07)
93 ENABLE W MATRIX INITIALIZATION
94
95 NO UPDATE OF EITHER STATE VECTOR (P20 OR P22)
96 INTERRUPT INTEGRATION AND GO TO P00
97 PERFORM ENGINE FAIL PROCEDURE
98
99 PLEASE ENABLE ENGINE

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1412THE

1

IN THE FOLLOWING NOUN LIST THE :NO LOAD: RESTRICTION MEANS THE NOUN
CONTAINS AT LEAST ONE COMPONENT WHICH CANNOT BE LOADED, I.E. OF
SCALE TYPE L (MIN/SEC), PP (2 INTEGERS) OR TT (LANDING RADAR POSITION).
IN THIS CASE VERBS 24 AND 25 ARE NOT ALLOWED, BUT VERBS 21, 22 OR 23
MAY BE USED TO LOAD ANY OF THE NOUN:S COMPONENTS WHICH ARE NOT OF THE
ABOVE SCALE TYPES.

THE :DEC ONLY: RESTRICTION MEANS ONLY DECIMAL OPERATION IS ALLOWED ON
EVERY COMPONENT IN THE NOUN. (NOTE THAT :NO LOAD: IMPLIES :DEC ONLY:.)

#	NORMAL NOUNS	COMPONENTS	SCALE AND DECIMAL POINT	RESTRICTIONS
# 00	NOT IN USE			
# 01	SPECIFY MACHINE ADDRESS (FRACTIONAL)	3COMP	.XXXXX FOR EACH	
# 02	SPECIFY MACHINE ADDRESS (WHOLE)	3COMP	XXXXX. FOR EACH	
# 03	SPECIFY MACHINE ADDRESS (DEGREES)	3COMP	XXX.XX DEG FOR EACH	
# 04	ANGULAR ERROR/DIFFERENCE	1COMP	XXX.XX DEG	
# 05	ANGULAR ERROR/DIFFERENCE	1COMP	XXX.XX DEG	
# 06	OPTION CODE	3COMP	OCTAL ONLY FOR EACH	
# 07	LOADING NOUN 07 WILL SET OR RESET SELECTED BITS IN ANY ERASABLE REGISTER ECADR OF WORD TO BE MODIFIED	3COMP	OCTAL ONLY FOR EACH	
#	ONES FOR BITS TO BE MODIFIED 1 TO SET OR 0 TO RESET SELECTED BITS			
# 08	ALARM DATA	3COMP	OCTAL ONLY FOR EACH	
# 09	ALARM CODES	3COMP	OCTAL ONLY FOR EACH	
# 10	CHANNEL TO BE SPECIFIED	1COMP	OCTAL ONLY	
# 11	TIG OF CSI	3COMP	00XXX. HRS	DEC ONLY MUST LOAD 3 COMPS
#			000XX. MIN	
#			0XX.XX SEC	
# 12	OPTION CODE (USED BY EXTENDED VERBS ONLY)	2COMP	OCTAL ONLY FOR EACH	
# 13	TIG OF CDH	3COMP	00XXX. HRS	DEC ONLY MUST LOAD 3 COMPS
#			000XX. MIN	
#			0XX.XX SEC	
# 14	CHECKLIST (USED BY EXTENDED VERBS ONLY) (NOUN 25 IS PASTED AFTER DISPLAY)	3COMP	XXXXX. FOR EACH	
# 15	INCREMENT MACHINE ADDRESS	1COMP	OCTAL ONLY	
# 16	TIME OF EVENT	3COMP	00XXX. HRS	DEC ONLY MUST LOAD 3 COMPS
#	(USED BY EXTENDED VERBS ONLY)		000XX. MIN	
#			0XX.XX SEC	
# 17	SPARE			
# 18	AUTO MANEUVER BALL ANGLES	3COMP	XXX.XX DEG FOR EACH	
# 19	SPARE			
# 20	ICDU ANGLES	3COMP	XXX.XX DEG FOR EACH	
# 21	PIPAS	3COMP	XXXXX. PULSES FOR EACH	
# 22	NEW ICDU ANGLES	3COMP	XXX.XX DEG FOR EACH	
# 23	SPARE			
# 24	DELTA TIME FOR AGC CLOCK	3COMP	00XXX. HRS	DEC ONLY MUST LOAD 3 COMPS
#			000XX. MIN	
#			0XX.XX SEC	

1412THE

#	MIXED NOUNS	COMPONENTS	SCALE AND DECIMAL POINT	RESTRICTIONS
# 40	TIME FROM IGNITION/CUTOFF	3COMP	XXBXX MIN/SEC	NO LOAD, DEC ONLY
#	VG,		XXXX.X FT/SEC	
#	DELTA V (ACCUMULATED)		XXXX.X FT/SEC	
# 41	TARGET AZIMUTH	2COMP	XXX.XX DEG	(FOR SYSTEM TEST)
#	ELEVATION		XX.XXX DEG	
# 42	APOGEE,	3COMP	XXXX.X NAUT MI	DEC ONLY
#	PERIGEE,		XXXX.X NAUT MI	
#	DELTA V (REQUIRED)		XXXX.X FT/SEC	
# 43	LATITUDE,	3COMP	XXX.XX DEG	DEC ONLY
#	LONGITUDE,		XXX.XX DEG	
#	ALTITUDE		XXXX.X NAUT MI	
# 44	APOGEE,	3COMP	XXXX.X NAUT MI	NO LOAD, DEC ONLY
#	PERIGEE,		XXXX.X NAUT MI	
#	TFF		XXBXX MIN/SEC	
# 45	MARKS,	3COMP	XXXXX.	NO LOAD, DEC ONLY
#	TFI OF NEXT BURN,		XXBXX MIN/SEC	
#	MGA		XXX.XX DEG	
# 46	AUTOPILOT CONFIGURATION	1COMP	OCTAL ONLY	
# 47	LEM WEIGHT,	2COMP	XXXXX. LBS	DEC ONLY
#	CSM WEIGHT		XXXXX. LBS	
# 48	GIMBAL PITCH TRIM,	2COMP	XXX.XX DEG	DEC ONLY
#	GIMBAL ROLL TRIM		XXX.XX DEG	
# 49	DELTA R,	3COMP	XXXX.X NAUT MI	DEC ONLY
#	DELTA V,		XXXX.X FT/SEC	
#	RADAR DATA SOURCE CODE		XXXXX.	
# 50	SPARE			
# 51	S-BAND ANTENNA ANGLES	2COMP	XXX.XX DEG	DEC ONLY
#	YAW		XXX.XX DEG	
# 52	CENTRAL ANGLE OF ACTIVE VEHICLE	1COMP	XXX.XX DEG	
# 53	SPARE			
# 54	RANGE,	3COMP	XXX.XX NAUT MI	DEC ONLY
#	RANGE RATE,		XXXX.X FT/SEC	
#	THETA		XXX.XX DEG	
# 55	NO. OF APSIDAL CROSSINGS	3COMP	XXXXX.	DEC ONLY
#	ELEVATION ANGLE		XXX.XX DEG	
#	CENTRAL ANGLE OF PASSIVE VEHICLE		XXX.XX DEG	
# 56	RR LOS AZIMUTH	2COMP	XXX.XX DEG	
#	ELEVATION		XXX.XX DEG	
# 57	DELTA R	1COMP	XXXX.X NAUT MI	DEC ONLY
# 58	PERIGEE ALT (POST TPI)	3COMP	XXXX.X NAUT MI	DEC ONLY
#	DELTA V TPI		XXXX.X FT/SEC	
#	DELTA V TPF		XXXX.X FT/SEC	
# 59	DELTA VELOCITY LOS	3COMP	XXXX.X FT/SEC FOR EA.	DEC ONLY
# 60	HORIZONTAL VELOCITY	3COMP	XXXX.X FT/SEC	DEC ONLY
#	ALTITUDE RATE		XXXX.X FT/SEC	
#	COMPUTED ALTITUDE		XXXXX. FEET	
# 61	TIME TO GO IN BRAKING PHASE	3COMP	XXBXX MIN/SEC	NO LOAD, DEC ONLY
#	TIME FROM IGNITION		XXBXX MIN/SEC	

#	CROSS RANGE DISTANCE		XXXX.X NAUT MI	
# 62	ABSOLUTE VALUE OF VELOCITY	3COMP	XXXX.X FT/SEC	NO LOAD, DEC ONLY
#	TIME FROM IGNITION		XXBXX MIN/SEC	
#	DELTA V (ACCUMULATED)		XXXX.X FT/SEC	
# 63	ABSOLUTE VALUE OF VELOCITY	3COMP	XXXX.X FT/SEC	DEC ONLY
#	ALTITUDE RATE		XXXX.X FT/SEC	
#	COMPUTED ALTITUDE		XXXXX. FEET	
# 64	TIME LEFT FOR REDESIGNATION- LPD ANGLE	3COMP	XXBXX	NO LOAD, DEC ONLY
#	ALTITUDE RATE		XXXX.X FT/SEC	
#	COMPUTED ALTITUDE		XXXXX. FEET	
# 65	SAMPLED AGC TIME	3COMP	00XXX. HRS.	DEC ONLY
#	(FETCHED IN INTERRUPT)		000XX. MIN	MUST LOAD 3 COMPS
#			0XX.XX SEC	
# 66	LR RANGE	2COMP	XXXXX. FEET	NO LOAD, DEC ONLY
#	POSITION		+0000X	
# 67	LRVX	3COMP	XXXXX. FT/SEC	
#	LRVY		XXXXX. FT/SEC	
#	LRVZ		XXXXX. FT/SEC	
# 68	SLANT RANGE TO LANDING SITE	3COMP	XXXX.X NAUT MI	NO LOAD, DEC ONLY
#	TIME TO GO IN BRAKING PHASE		XXBXX MIN/SEC	
#	LR ALTITUDE - COMPUTED ALTITUDE		XXXXX. FEET	
# 69	SPARE			
# 70	AOT DETENT CODE/STAR CODE	3COMP	OCTAL ONLY FOR EACH	
# 71	AOT DETENT CODE/STAR CODE	3COMP	OCTAL ONLY FOR EACH	
# 72	RR 360 - TRUNNION ANGLE	2COMP	XXX.XX DEG	
#	SHAFT ANGLE		XXX.XX DEG	
# 73	NEW RR 360 - TRUNNION ANGLE	2COMP	XXX.XX DEG	
#	SHAFT ANGLE		XXX.XX DEG	
# 74	TIME FROM IGNITION	3COMP	XXBXX MIN/SEC	NO LOAD, DEC ONLY
#	YAW AFTER VEHICLE RISE		XXX.XX DEG	
#	PITCH AFTER VEHICLE RISE		XXX.XX DEG	
# 75	DELTA ALTITUDE CDH	3COMP	XXXX.X NAUT MI	NO LOAD, DEC ONLY
#	DELTA TIME (CDH-CSI OR TPI-CDH)		XXBXX MIN/SEC	
#	DELTA TIME (TPI-CDH OR TPI-NOMTPI)		XXBXX MIN/SEC	
# 76	DESIRED HORIZONTAL VELOCITY	3COMP	XXXX.X FT/SEC	DEC ONLY
#	DESIRED RADIAL VELOCITY		XXXX.X FT/SEC	
#	CROSS-RANGE DISTANCE		XXXX.X NAUT MI	
# 77	TIME TO ENGINE CUTOFF	2COMP	XXBXX MIN/SEC	NO LOAD, DEC ONLY
#	VELOCITY NORMAL TO CSM PLANE		XXXX.X FT/SEC	
# 78	RR RANGE	2COMP	XXX.XX NAUT MI	NO LOAD, DEC ONLY
#	RANGE RATE		XXXXX. FT/SEC	
# 79	CURSOR ANGLE	3COMP	XXX.XX DEG	DEC ONLY
#	SPIRAL ANGLE		XXX.XX DEG	
#	POSITION CODE		XXXXX.	
# 80	DATA INDICATOR,	2COMP	XXXXX.	
#	OMEGA		XXX.XX DEG	
# 81	DELTA V (LV)	3COMP	XXXX.X FT/SEC FOR EACH	DEC ONLY

14121HE

ASSEMBLY_AND_OPERATION_INFORMATION

REGISTERS AND SCALING FOR NORMAL NOUNS

# NOUN	REGISTER	SCALE TYPE
# 00	NOT IN USE	
# 01	SPECIFY ADDRESS	B
# 02	SPECIFY ADDRESS	C
# 03	SPECIFY ADDRESS	D
# 04	DSPTM1	H
# 05	DSPTM1	H
# 06	OPTION1	A
# 07	XREG	A
# 08	ALMCADR	A
# 09	FAILREG	A
# 10	SPECIFY CHANNEL	A
# 11	TCSI	K
# 12	OPTIONX	A
# 13	TCDH	K
# 14	DSPTMX	C
# 15	INCREMENT ADDRESS	A
# 16	DSPTMX	K
# 17	SPARE	
# 18	FDAIX	D
# 19	SPARE	
# 20	CDUX	D
# 21	PIPAX	C
# 22	THETAD	D
# 23	SPARE	
# 24	DSPTM2 +1	K
# 25	DSPTM1	C
# 26	DSPTM1	A
# 27	SMODE	C
# 28	SPARE	
# 29	SPARE	
# 30	SPARE	
# 31	SPARE	
# 32	-TPER	K
# 33	TIG	K
# 34	DSPTM1	K
# 35	TTOGO	K
# 36	TIME2	K
# 37	TTPI	K
# 38	TET	K
# 39	SPARE	

ASSEMBLY_AND_OPERATION_INFORMATION

REGISTERS AND SCALING FOR MIXED NOUNS

# NOUN	COMP	REGISTER	SCALE TYPE
# 40	1	TTOGO	L
#	2	VGDISP	S
#	3	DVTOTAL	S
# 41	1	DSPTM1	D
#	2	DSPTM1 +1	E
# 42	1	HAPO	Q
#	2	HPER	Q
#	3	VGDISP	S
# 43	1	LAT	H
#	2	LONG	H
#	3	ALT	Q
# 44	1	HAPOX	Q
#	2	HPERX	Q
#	3	TFF	L
# 45	1	TRKMKCNT	C
#	2	TTOGO	L
#	3	+MGA	H
# 46	1	DAPDATR1	A
# 47	1	LEMMASS	KK
#	2	CSMMASS	KK
# 48	1	PITTIME	NN
#	2	ROLLTIME	NN
# 49	1	R22DISP	Q
#	2	R22DISP +2	S
#	3	WHCHREAD	C
# 50	SPARE		
# 51	1	ALPHASB	H
#	2	BETASB	H
# 52	1	ACTCENT	H
# 53	SPARE		
# 54	1	RANGE	JJ
#	2	RRATE	S
#	3	RTHETA	H
# 55	1	NN	C
#	2	ELEV	H
#	3	CENTANG	H
# 56	1	RR-AZ	H
#	2	RR-ELEV	H
# 57	1	DELTAR	Q
# 58	1	POSTTPI	Q
#	2	DELVTPI	S
#	3	DELVTPI	S
# 59	1	DVLOS	S
#	2	DVLOS +2	S
#	3	DVLOS +4	S
# 60	1	VHORIZ	S

#	2	HDOTDISP	S
#	3	HCALC	RR
# 61	1	TTFDISP	L
#	2	TTOGO	L
#	3	OUTOFPLN	QQ
# 62	1	ABVEL	S
#	2	TTOGO	L
#	3	DVTOTAL	S
# 63	1	ABVEL	S
#	2	HDOTDISP	S
#	3	HCALC1	RR
# 64	1	FUNNYDSP	PP
#	2	HDOTDISP	S
#	3	HCALC	RR
# 65	1	SAMPTIME	K
#	2	SAMPTIME	K
#	3	SAMPTIME	K
# 66	1	RSTACK +6	W
#	2	CHANNEL 33	TT
# 67	1	RSTACK	X
#	2	RSTACK +2	Y
#	3	RSTACK +4	Z
# 68	1	RANGEDSP	QQ
#	2	TTFDISP	L
#	3	DELTAH	RR
# 69	SPARE		
# 70	1	AOTCODE	A
#	2	AOTCODE +1	A
#	3	AOTCODE +2	A
# 71	1	AOTCODE	A
#	2	AOTCODE +1	A
#	3	AOTCODE +2	A
# 72	1	CDUT	WW
#	2	CDUS	D
# 73	1	TANG	WW
#	2	TANG +1	D
# 74	1	TTOGO	L
#	2	YAW	H
#	3	PITCH	H
# 75	1	DIFFALT	Q
#	2	T1TOT2	L
#	3	T2TOT3	L
# 76	1	ZDOTD	S
#	2	RDOTD	S
#	3	XRANGE	Q
# 77	1	TTOGO	L
#	2	YDOT	S
# 78	1	RSTACK	U
#	2	RSTACK +2	V
# 79	1	CURSOR	D

ASSEMBLY_AND_OPERATION_INFORMATION

#	2	SPIRAL	D
#	3	POSCODE	C
# 80	1	DATAGOOD	C
#	2	OMEGAD	H
# 81	1	DEVLVLC	S
#	2	DEVLVLC +2	S
#	3	DEVLVLC +4	S
# 82	1	DEVLVLC	S
#	2	DEVLVLC +2	S
#	3	DEVLVLC +4	S
# 83	1	DELVIMU	S
#	2	DELVIMU +2	S
#	3	DELVIMU +4	S
# 84	1	DELVOV	S
#	2	DELVOV +2	S
#	3	DELVOV +4	S
# 85	1	VGBODY	S
#	2	VGBODY +2	S
#	3	VGBODY +4	S
# 86	1	DEVLVLC	S
#	2	DEVLVLC +2	S
#	3	DEVLVLC +4	S
# 87	1	AZ	D
#	2	EL	D
# 88	1	STARAD	B
#	2	STARAD +2	B
#	3	STARAD +4	B
# 89	1	LANDLAT	G
#	2	LANDLONG	G
#	3	LANDALT	JJ
# 90	1	RANGE	JJ
#	2	RRATE	S
#	3	RTHETA	H
# 91	1	P21ALT	Q (MEMORY/100 TO DISPLAY TENS N.M.)
#	2	P21VEL	P
#	3	P21GAM	H
# 92	SPARE		
# 93	1	OGC	G
#	2	OGC +2	G
#	3	OGC +4	G
# 94	SPARE		
# 95	SPARE		
# 96	SPARE		
# 97	1	DSPTM1	C
#	2	DSPTM1 +1	C
#	3	DSPTM1 +2	C
# 98	1	DSPTM2	C
#	2	DSPTM2 +1	B
#	3	DSPTM2 +2	C
# 99	1	WWPOS	XX

ASSEMBLY AND OPERATION INFORMATION

NOUN SCALES AND FORMATS

# -SCALE TYPE-	PRECISION		
# UNITS	DECIMAL FORMAT	--	AGC FORMAT
# -----	-----	--	-----
# -A-			
# OCTAL	XXXXX	SP	OCTAL
# -B-			
# FRACTIONAL	.XXXXX	SP	⁻¹⁴ BIT 1 = 2 UNITS
#	(MAX .99996)		
# -C-			
# WHOLE	XXXXX.	SP	BIT 1 = 1 UNIT
#	(MAX 16383.)		
# -D-			
# CDU DEGREES	XXX.XX DEGREES	SP	¹⁵ BIT 1 = 360/2 DEGREES
#	(MAX 359.99)		(USES 15 BITS FOR MAGNI-
#			TUDE AND 2-S COMP.)
# -E-			
# ELEVATION DEGREES	XX.XXX DEGREES	SP	¹⁴ BIT 1 = 90/2 DEGREES
#	(MAX 89.999)		
# -F-			
# DEGREES (180)	XXX.XX DEGREES	SP	¹⁴ BIT 1 = 180/2 DEGREES
#	(MAX 179.99)		
# -G-			
# DP DEGREES(90)	XX.XXX DEGREES	DP	BIT 1 OF LOW REGISTER =
#			²⁸
#			360/2 DEGREES
# -H-			
# DP DEGREES (360)	XXX.XX DEGREES	DP	BIT 1 OF LOW REGISTER =
#			²⁸
#	(MAX 359.99)		360/2 DEGREES

# -K-				
# TIME (HR, MIN, SEC)	00XXX. HR	DP	BIT 1 OF LOW REGISTER =	
#	000XX. MIN		-2	
#	0XX.XX SEC		10 SEC	
#	(DECIMAL ONLY.			
#	MAX MIN COMP=59			
#	MAX SEC COMP=59.99			
#	MAX CAPACITY=745 HRS			
#	39 MINS			
#	14.55 SECS.			
#	WHEN LOADING, ALL 3			
#	COMPONENTS MUST BE			
#	SUPPLIED.)			
# -L-				
# TIME (MIN/SEC)	XXBXX MIN/SEC	DP	BIT 1 OF LOW REGISTER =	
#	(B IS A BLANK		-2	
#	POSITION, DECIMAL		10 SEC	
#	ONLY, DISPLAY OR			
#	MONITOR ONLY. CANNOT			
#	BE LOADED.			
#	MAX MIN COMP=59			
#	MAX SEC COMP=59			
#	VALUES GREATER THAN			
#	59 MIN 59 SEC			
#	ARE DISPLAYED AS			
#	59 MIN 59 SEC.)			
# -M-			-2	
# TIME (SEC)	XXX.XX SEC	SP	BIT 1 = 10 SEC	
#	(MAX 163.83)			
# -N-				
# TIME(SEC) DP	XXX.XX SEC	DP	BIT 1 OF LOW REGISTER =	
#			-2	
#			10 SEC.	
# -P-				
# VELOCITY 2	XXXXX. FEET/SEC	DP	BIT 1 OF HIGH REGISTER =	
#	(MAX 41994.)		-7	
#			2 METERS/CENTI-SEC	
# -Q-				
# POSITION 4	XXXX.X NAUTICAL MILES	DP	BIT 1 OF LOW REGISTER =	
#			2 METERS	
# -S-				
# VELOCITY 3	XXXX.X FT/SEC	DP	BIT 1 OF HIGH REGISTER =	
#			-7	
#			2 METERS/CENTI-SEC	

# -T-				
# G	XXX.XX G	SP	BIT 1 = 10 ⁻² G	
#	(MAX 163.83)			
# -U-				
# RENDEZVOUS	XXX.XX NAUT MI	DP	LOW ORDER BIT OF LOW ORDER	
# RADAR RANGE			WORD = 9.38 FEET	
# -V-				
# RENDEZVOUS	XXXXX. FEET/SEC	DP	LOW ORDER BIT OF LOW ORDER	
# RADAR RANGE RATE			WORD = -.6278 FEET/SEC	
# -W-				
# LANDING RADAR	XXXXX. FEET	DP	LOW ORDER BIT OF LOW ORDER	
# ALTITUDE			WORD = 1.079 FEET	
# -X-				
# LANDING RADAR	XXXXX. FEET/SEC	DP	LOW ORDER BIT OF LOW ORDER	
# VELX			WORD = -.6440 FEET/SEC	
# -Y-				
# LANDING RADAR	XXXXX. FEET/SEC	DP	LOW ORDER BIT OF LOW ORDER	
# VELY			WORD = 1.212 FEET/SEC	
# -Z-				
# LANDING RADAR	XXXXX. FEET/SEC	DP	LOW ORDER BIT OF LOW ORDER	
# VELZ			WORD = .8668 FEET/SEC	
# -AA-				
# INITIAL/FINAL	XXXXX. FEET	DP	LOW ORDER BIT OF LOW ORDER	
# ALTITUDE			WORD = 2.345 FEET	
# -BB-				
# ALTITUDE RATE	XXXXX. FEET/SEC	SP	LOW ORDER BIT = .5	
#	(MAX 08191.)		FEET/SEC	
# -CC-				
# FORWARD/LATERAL	XXXXX. FEET/SEC	SP	LOW ORDER BIT = .5571	
# VELOCITY	(MAX 09126.)		FEET/SEC	
# -DD-				
# ROTATIONAL HAND	XXXXX. DEG/SEC	SP	FRACTIONAL PART OF PI RAD	
# CONTROLLER ANGULAR	(MAX 00044.)		4 SEC	
# RATES				
# -EE-				
# OPTICAL TRACKER	XXX.XX DEG.	DP	LOW ORDER BIT OF LOW ORDER	
# AZIMUTH ANGLE			15	
#			WORD = 360/2 DEGREES	

-JJ-

POSITION5 XXX.XX NAUT MI DP BIT 1 OF LOW REGISTER =
2 METERS

-KK-

WEIGHT2 XXXXX. LBS SP FRACTIONAL PART OF 2^{16} KG

-NN-

TRIM DEGREES 2 XXX.XX DEG SP BIT 1=.01 SEC(TIME)
(MAX 032.76)

-PP-

2 INTEGERS +XXBYY DP BIT 1 OF HIGH REGISTER =
(B IS A BLANK 1 UNIT OF XX
POSITION. DECIMAL BIT 1 OF LOW REGISTER =
ONLY, DISPLAY OR 1 UNIT OF YY
MONITOR ONLY. CANNOT (EACH REGISTER MUST
BE LOADED.) CONTAIN A POSITIVE INTEGER
(MAX 99B99) LESS THAN 100)

-QQ-

POSITION7 XXXX.X NAUT MI DP BIT 1 OF LOW REGISTER =
(MAX 9058.9) -4
2 METERS

-RR-

COMPUTED ALTITUDE XXXXX. FEET DP BIT 1 OF LOW REGISTER =
-4
2 METERS

-SS-

DP DEGREES XXXX.X DEGREES DP BIT 1 OF HIGH REGISTER =
1 DEGREE

-TT-

LANDING RADAR +0000X CHANNEL 33, BIT 6 = NOT POSIT. 1
POSITION (DECIMAL ONLY. CHANNEL 33, BIT 7 = NOT POSIT. 2
DISPLAY OR MONITOR X = 1 FOR LR POSITION 1
ONLY. CANNOT BE X = 2 FOR LR POSITION 2
LOADED.)

-WW-

360-CDU DEGREES XXX.XX DEGREES SP BIT 1 = $360 - (360/2^{15})$
(MAX 359.99) DEGREES
(USES 15 BITS FOR MAGNI-
TUDE AND 2-S COMP.)

-XX-

POSITION 9 XXXXX. FEET DP BIT 1 OF LOW REGISTER =
-9
2 METERS



ASSEMBLY_AND_OPERATION_INFORMATION

-YY-

VELOCITY 4

XXXX.X FEET/SEC

DP

FRACTIONAL PART

#

(MAX 328.0)

METERS/CENTI-SEC

-AAA-

RADIANS

XX.XXX RADIANS

DP

BIT 1 OF HIGH REGISTER =

#

(MAX 31.999)

-9

#

2 RADIANS.

THAT-S ALL ON THE NOUNS.

ALARM CODES FOR LUMINARY

#	*9	*18	*60	COLUMN
#	CODE	* TYPE	SET BY	
#	00105	** AOTMARK SYSTEM IN USE		
#	00107	MORE THAN 5 MARK PAIRS	AOTMARK	
#	00111	MARK MISSING	AOTMARK	
#	00112	MARK OR MARK REJECT NOT BEING ACCEPTED	AOTMARK	
#	00113	NO INBITS	AOTMARK	
#	00114	MARK MADE BUT NOT DESIRED	AOTMARK	
#	00115	NO MARKS IN LAST PAIR TO REJECT	AOTMARK	
#	00206	ZERO ENCODE NOT ALLOWED WITH COARSE ALIGN	IMU MODE SWITCHING	
#	00206	+ GIMBAL LOC		
#	00207	ISS TURNON REQUEST NOT PRESENT FOR 90 SEC	T4RUPT	
#	00210	IMU NOT OPERATING	IMU MODE SWITCH, IMU-2, RD2, P51, P57	
#	00211	COARSE ALIGN ERROR	IMU MODE SWITCH	
#	00212	PIPA FAIL BUT PIPA IS NOT BEING USED	IMU MODE SWITCH, T4RPT	
#	00213	IMU NOT OPERATING WITH TURN-ON REQUEST	T4RUPT	
#	00214	PROGRAM USING IMU WHEN TURNED OFF	T4RUPT	
#	00217	BAD RETURN FROM IMUSTALL	P51, P52, P57	
#	00220	IMU NOT ALIGNED - NO REFSMMAT	R02, R47	
#	00401	DESIRED GIMBAL ANGLE YIELDS GIMBAL LOCK	INF ALIGN, IMU-2,	
#			FINDCDUW	
#	00402	FINDCDUW NOT CONTROLLING ATTITUDE	FINDCDUW	
#	00404	TWO STARS NOT AVAILABLE IN ANY DETENT	R59, LUNAR SURFACE	
#	00405	TWO STARS NOT AVAILABLE	P52	
#	00421	W-MATRIX OVERFLOW	INTEGRV	
#	00430	** ACCELERATION OVERFLOW IN INTEGRATION	ORBITAL INTEGRATION	
#	00501	P RADAR ANTENNA OUT OF LIMITS	R23	
#	00502	BAD RADAR GIMBAL ANGLE INPUT	V41N72	
#	00503	P RADAR ANTENNA DESIGNATE FAIL	R21, NON-P IN V41N72	
#	00510	RADAR AUTO DESCRETE NOT PRESENT	R25	
#	00511	LR NOT IN POSITION 2 OR REPOSITIONING	SERVICER	
#	00514	P RR GOES OUT OF AUTO MODE WHILE IN USE	P20	
#	00515	RR CDU FAIL DISCRETE PRESENT	R25	
#	00520	RADAR RUPT NOT EXPECTED AT THIS TIME	RADAR READ	
#	00521	COULD NOT READ RADAR	P20	
#	00522	LANDING RADAR POSITION CHANGE	RADAR READ	
#	00523	P LR ANTENNA DIDN'T ACHIEVE POSITION 2	SERVICER, V60 (NON-P IN V60)	
#	00525	P DELTA THETA GREATER THAN 3 DEGREES	R22	
#	00526	P RANGE GREATER THAN 400 NAUT. MILES	P20, P22	
#	00527	P LOS NOT IN MODE II COVERAGE WHILE ON	R21, R24	
#		LUNAR SURFACE		
#		OR VEHICLE MANEUVER REQUIRED	R24 (20)	
#	00530	P LOS NOT IN MODE2 COVERAGE	R21	
#		ON LUNAR SURFACE AFTER 600 SECS.		
#	00600	IMAGINARY ROOTS ON FIRST ITERATION	P32, P72	
#	00601	PERIGEE ALTITUDE CSI LT PMIN1	P32, P72.	

# 00602	PERIGEE ALTITUDE CDH LT PMIN2	P32, P72.
# 00603	CSI TO CDH TIME LT TMIN12	P32, P72, P33, P73
# 00604	CDH TO TPI TIME LT TMIN23	P32, P72,
#	OR COMPUTED CDH TIME GREATER THAN INPUT TP1 TIME	
# 00605	NUMBER OF ITERATIONS EXCEEDS LOOP MAXIMUM	P32, P72
# 00606	DV EXCEEDS MAXIMUM	P32, P72
# 00607	** NO SOLN FROM TIME-THETA OR TIME-RADIUS	TIMETHET, TIMERAD
# 00611	NO TIG FOR GIVEN ELEV ANGLE	P34, P74
# 00701	ILLEGAL OPTION CODE SELECTED	P57
# 00777	PIPA FAIL CAUSED THE ISS WARNING	T4RUPT
# 01102	AGC SELF TEST ERROR	SELF CHECK
# 01103	** UNUSED CCS BRANCH EXECUTED	ABORT
# 01104	* DELAY ROUTINE BUSY	EXEC
# 01105	DOWNLINK TOO FAST	T4RUPT
# 01106	UPLINK TOO FAST	T4RUPT
# 01107	PHASE TABLE FAILURE. ASSUME	RESTART
#	ERASABLE MEMORY IS SUSPECT.	RESTART
# 01201	* EXECUTIVE OVERFLOW - NO VAC AREAS	EXEC
# 01202	* EXECUTIVE OVERFLOW - NO CORE SETS	EXEC
# 01203	* WAITLIST OVERFLOW - TOO MANY TASKS	WAITLIST
# 01204	** WAITLIST, VARDELAY, FIXDELAY, OR LONGCALL	WAITLIST ROUTINES
#	CALLED WITH ZERO OR NEGATIVE DELTA-TIME	
# 01206	** SECOND JOB ATTEMPTS TO GO TO SLEEP	PINBALL
# 01206	VIA KYBD AND DISPLAY PROGRAM	
# 01207	* NO VAC AREAS FOR MARKS	AOTMARK
# 01210	* TWO PROGRAMS USING DEVICE AT SAME TIME	MODE SWITCHING
# 01211	* ILLEGAL INTERRUPT OF EXTENDED VERB	AOTMARK
# 01301	ARCSIN-ARCCOS ARGUMENT TOO LARGE	INTERPRETER
# 01302	** SQRT CALLED WITH NEGATIVE ARGUMENT	INTERPRETER
# 01406	BAD RETURN FROM ROOTPSRS	DESCENT GUIDANCE EQS.
# 01406	** BAD RETURN FROM ROOTPSRS	IGNITION ALGORITHM
#	NOTE: 1406 IS A POODOO DURING THE IGNITION ALGORITHM	
#	AND AN ALARM DURING THE ACTUAL GUIDANCE PHASE.	
#		
# 01407	VG INCREASING (DELTA-V ACCUMULATED	S40.8
#	.GT. 90 DEGREES AWAY FROM DESIRED THRUST	S40.8
#	VECTOR.)	S40.8
# 01410	UNINTENTIONAL OVERFLOW IN GUIDANCE	DESCENT GUIDANCE EQS.
# 01412	DESCENT IGNALG NOT CONVERGING	P63
# 01501	** KEYBOARD AND DISPLAY ALARM DURING	PINBALL
# 01501	INTERNAL USE (NVSUB). ABORT	
# 01502	** ILLEGAL FLASHING DISPLAY	GOPLAY
# 01520	V37 REQUEST NOT PERMITTED AT THIS TIME	V37
# 01600	OVERFLOW IN DRIFT TEST	IMU 4
# 01601	BAD IMU TORQUE	OPT PRE ALIGN CALIB
# 01601		IMU 4 (LEM)
# 01703	IGNITION TIME SLIPPED	MIDTOAVE
# 01706	INCORRECT PROGRAM REQUESTED FOR VEHICLE	
#	CONFIGURATION	P40, P42

# 02000	* DAP STILL IN PROGRESS AT NEXT TIMES RUPT	DAP
# 02001	JET FAILURES HAVE DISABLED Y-Z TRANS.	DAP
# 02002	JET FAILURES HAVE DISABLED X TRANSLATION	DAP
# 02003	JET FAILURES HAVE DISABLED P-ROTATION	DAP
# 02004	JET FAILURES HAVE DISABLED U-V ROTATION	DAP
# 03777	ICDU FAIL CAUSED THE ISS WARNING	T4RUPT
# 04777	ICDU, PIPA FAILS CAUSED THE ISS WARNING	T4RUPT
# 07777	IMU FAIL CAUSED THE ISS WARNING	T4RUPT
# 10777	IMU, PIPA FAILS CAUSED THE ISS WARNING	T4RUPT
# 13777	IMU, ICDU FAILS CAUSED THE ISS WARNING	T4RUPT
# 14777	IMU, ICDU, PIPA FAILS CAUSED THE ISS WARNING	T4RUPT
#		
#	* INDICATES AN ABORT CODE THAT RESULTS IN A SOFTWARE RESTART.	
#		
#	** INDICATES A MORE SERIOUS ABORT CODE THAT RESULTS IN THE	
#	PROGRAM GOING TO R00.	
#		
#	P INDICATES A PRIORITY ALARM.	
#		
#	ALL OTHERS ARE NON-ABORTIVE	

CHECKLIST CODES FOR LUMINARY

# *9	*17	*26	*9 COLUMN
# R1CODE		ACTION TO BE EFFECTED	PROGRAM
# 00013	KEY IN	NORMAL OR GYRO TORQUE COARSE ALIGN	P52
# 00014	PROCEED	DO IMU FINE ALIGN ROUTINE	P51, P63, P57
# 00014	ENTER	DO LANDING SITE DETERMINATION (N89DISP)	P57OPTION2
# 00015	PERFORM	CELESTIAL BODY ACQUISITION	R51, P51
# 00062	SWITCH	AGC POWER DOWN	P06
# 00201	SWITCH	RR MODE TO AUTOMATIC	P20, P22, R04
# 00203	SWITCH	GUID CONTROL TO GNC, MODE TO AUTO...	P12, P42, P71
#		ALSO THR CONT TO AUTO	P40, P63, P70
# 00205	PERFORM	MANUAL ACQUISITION OF RR	R23
# 00500	SWITCH	LR ANTENNA TO POSITION 1	P63

SWITCH DENOTES CHANGE POSITION OF A CONSOLE SWITCH
PERFORM DENOTES START OR END OF A TASK
KEY IN DENOTES KEY IN OF DATA THRU THE DSKY

OPTION CODES FOR LUMINARY

THE SPECIFIED OPTION CODES WIL BE FLASHED IN COMPONENT R1 IN
CONJUNCTION WITH V04N06 OR V04N12 (FOR EXTENDED VERBS) TO REQUEST THE
ASTRONAUT TO LOAD INTO COMPONENT R2 THE OPTION HE DESIRES.

# *9	*17	*52	*11	*25	COLUMN
# OPTION # CODE	PURPOSE	INPUT FOR COMPONENT 2	PROGRAM(S)	APPLICABILITY	
# 00001	SPECIFY IMU ORIENTATION	1=PREF 2=NOM 3=REFSMMAT 4=LAND SITE	P52	ALL	
# 00002	SPECIFY VEHICLE	1=THIS 2=OTHER	P21,R30	ALL	
# 00003	SPECIFY TRACKING ATTITUDE	1=PREFERRED 2=OTHER	R63	ALL	
# 00004	SPECIFY RADAR	1=RR 2=LR	R04	SUNDANCE + LUMINARY	
# 00005	SPECIFY SOR PHASE	1=FIRST 2=SECOND	P38	COLOSSUS + LUMINARY	
# 00006	SPECIFY RR COARSE ALIGN OPTION	1=LOCKON 2=CONTINUOUS DESIG.	V41N72	SUNDANCE + LUMINARY	
# 00010	SPECIFY ALIGNMENT MODE	0=ANY TIME 1=REFSMMAT +G 2=TWO BODIES 3=ONE BODY + G	P57	LUMINARY	
# 00012	SPECIFY CSM ORBIT OPTION	1=NO ORBIT CHANGE 2=CHANGE ORBIT TO PASS OVER LM	P22	LUMINARY	

TAGS_FOR_RELATIVE_SETLOC

TAGS FOR RELATIVE SETLOC AND BLANK BANK CARDS

COUNT BANKSUM

MODULE 1 CONTAINS BANKS 0 THROUGH 5

RADARFF BLOCK 02

FFTAG1 EQUALS

FFTAG2 EQUALS

FFTAG3 EQUALS

FFTAG4 EQUALS

FFTAG7 EQUALS

FFTAG8 EQUALS

FFTAG9 EQUALS

FFTAG10 EQUALS

FFTAG11 EQUALS

FFTAG12 EQUALS

FFTAG13 EQUALS

BNKSUM 02

FFTAG5 BLOCK 03

FFTAG6 EQUALS

BNKSUM 03

DLAYJOB BANK 00

EQUALS

BNKSUM 00

RESTART BANK 01

EQUALS

LOADDAP1 EQUALS

BNKSUM 01

R02 BANK 04

VERB37 EQUALS

PINBALL4 EQUALS

CONICS1 EQUALS

KEYRUPT EQUALS

R36LM EQUALS

UPDATE2 EQUALS

E/PROG EQUALS

AOTMARK2 EQUALS

BNKSUM 04

1412THE

1	# PAGE 1 FOR ALTERNATIVE LEVELS			1
2	INTPRET2	EQUALS		2
3		BNKSUM	12	3
4				4
5		BANK	13	5
6	LATLONG	EQUALS		6
7	INTINIT	EQUALS		7
8	LEMGEOM	EQUALS		8
9	P76LOC	EQUALS		9
10	ORBITAL2	EQUALS		10
11	ABTFLGS	EQUALS		11
12		BNKSUM	13	12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

TAGS_FOR_RELATIVE_SETLOC

MODULE 3 CONTAINS BANKS 14 THROUGH 21

P50S1	BANK	14
STARTAB	EQUALS	
ASENT4	EQUALS	
	BNKSUM	14

P50S	BANK	15
EPHEM	EQUALS	
	BNKSUM	15

DAPS1	BANK	16
	EQUALS	
	BNKSUM	16

DAPS2	BANK	17
P40S3	EQUALS	
	EQUALS	
	BNKSUM	17

DAPS3	BANK	20
LOADDAP	EQUALS	
RODTRAP	EQUALS	
	BNKSUM	20

DAPS4	BANK	21
R10	EQUALS	
R11	EQUALS	
	BNKSUM	21

MODULE 4 CONTAINS BANKS 22 THROUGH 27

BANK 22
KALCMON1 EQUALS
KALCMON2 EQUALS
R30LOC EQUALS
RENDEZ EQUALS
SERV2 EQUALS
LANDCNST EQUALS
BNKSUM 22

BANK 23
POWFLITE EQUALS
POWFLIT1 EQUALS
INFLIGHT EQUALS
AOPERI EQUALS
R61 EQUALS
R62 EQUALS
INTPRET1 EQUALS
MEASINC EQUALS
MEASINC1 EQUALS
EXTVB1 EQUALS
P12A EQUALS
NORMLIZ EQUALS
ASENT7 EQUALS
BNKSUM 23

BANK 24
PLANTIN EQUALS
P20S EQUALS
BNKSUM 24

BANK 25
P20S1 EQUALS
P20S2 EQUALS
RADARUPT EQUALS
RRLEADIN EQUALS
R29S1 EQUALS
PLANTIN3 EQUALS
BNKSUM 25

BANK 26

TAGS_FOR_RELATIVE_SETLOC

MODULE 5 CONTAINS BANKS 30 THROUGH 35

LOWSUPER BANK 30
P12 EQUALS
ASENT EQUALS
FCDUW EQUALS
FLOGSUB EQUALS
VB67A EQUALS
ASENT5 EQUALS
BNKSUM 30

FTHROT BANK 31
F2DPS*31 EQUALS
VB67 EQUALS
BNKSUM 31

P20S4 BANK 32
F2DPS*32 EQUALS
ABORTS EQUALS
LRS22 EQUALS
P66LOC EQUALS
R47 EQUALS
SERV EQUALS
BNKSUM 32

SERVICES BANK 33
R29/SERV EQUALS
ASENT6 EQUALS
BNKSUM 33

STBLEORB BANK 34
P30S1 EQUALS
CSI/CDH1 EQUALS
ASCFILT EQUALS
R12STUFF EQUALS
SERV4 EQUALS
BNKSUM 34

CSI/CDH

BANK 35

BANK EQUALS

P30S

EQUALS

GLM

EQUALS

P40S2

EQUALS

BNKSUM	35
--------	----

TAGS_FOR_RELATIVE_SETLOC

MODULE 6 CONTAINS BANKS 36 THROUGH 43

P40S BANK 36
EQUALS
BNKSUM 36

P05P06 BANK 37
EQUALS
IMU2 EQUALS
IMU4 EQUALS
R31 EQUALS

IMUSUPER EQUALS
SERV1 EQUALS
BNKSUM 37

PINBALL1 BANK 40
EQUALS
SELSUPR EQUALS
PINSUPER EQUALS
R31LOC EQUALS
BNKSUM 40

PINBALL2 BANK 41
EQUALS
BNKSUM 41

SBAND BANK 42
PINBALL3 EQUALS
EQUALS
BNKSUM 42

EXTVERBS BANK 43
SELFCHC EQUALS
BNKSUM 43

```
1
2
3      HI6ZEROS      EQUALS  ZEROVECS      # ZERO VECTOR ALWAYS IN HIGH MEMORY
4      LO6ZEROS      EQUALS  ZEROVEC       # ZERO VECTOR ALWAYS IN LOW MEMORY
5      HIDPHALF      EQUALS  UNITX
6      LODPHALF      EQUALS  XUNIT
7      HIDP1/4       EQUALS  DP1/4TH
8      LODP1/4       EQUALS  D1/4          # 2DEC .25
9      HIUNITX       EQUALS  UNITX
10     HIUNITY       EQUALS  UNITY
11     HIUNITZ       EQUALS  UNITZ
12     LOUNITX       EQUALS  XUNIT          # 2DEC .5
13     LOUNITY       EQUALS  YUNIT          # 2DEC 0
14     LOUNITZ       EQUALS  ZUNIT          # 2DEC 0
```

```
15
16     DELRSPL       EQUALS  SPLRET          # COL PGM, ALSO CALLED BY R30 IN LUMINARY.
```

```
17
18     # ROPE-SPECIFIC ASSIGNS OBVIATING NEED TO CHECK COMPUTER FLAG IN DETERMINING INTEGRATION AREA ENTRIES.
```

```
19
20     ATOPTHIS      EQUALS  ATOPLEM
21     ATOPOTH       EQUALS  ATOPCSM
22     OTHPREC       EQUALS  CSMPREC
23     MOONTHIS      EQUALS  LMOONFLG
24     MOONOTH       EQUALS  CMOONFLG
25     MOVATHIS      EQUALS  MOVEALEM
26     RMM           =      LODPMAX
27     RME           =      LODPMAX1
28     THISPREC      EQUALS  LEMPREC
29     THISAXIS      =      UNITZ
30     NB1NB2        EQUALS  THISAXIS      # FOR R31
31     ERASID        EQUALS  BITS2-10      # DOWNLINK ERASABLE DUMP ID
32     DELAYNUM      EQUALS  TWO
```

DPS AND APS ENGINE PARAMETERS

SETLOC P40S
BANK
COUNT* \$\$/P40

*** THE ORDER OF THE FOLLOWING SIX CONSTANTS MUST NOT BE CHANGED ***

FDPS	2DEC	4.3670 B-7	# 9817.5 LBS FORCE IN NEWTONS
MDOTDPS	2DEC	0.1480 B-3	# 32.62 LBS/SEC IN KGS/CS
DTDECAY	2DEC	-38	
FAPS	2DEC	1.5569 B-7	# 3500 LBS FORCE IN NEWTONS
MDOTAPS	2DEC	0.05135 B-3	# 11.32 LBS/SEC IN KGS/CS
ATDECAY	2DEC	-10	

FRCS4	2DEC	0.17792 B-7	# 400 LBS FORCE IN NEWTONS
FRCS2	2DEC	0.08896 B-7	# 200 LBS FORCE IN NEWTONS

SETLOC P40S1
BANK
COUNT* \$\$/P40

*** APS IMPULSE DATA FOR P42 *****

K1VAL	2DEC	124.55 B-23	# 2800 LB-SEC
K2VAL	2DEC	31.138 B-24	# 700 LB-SEC
K3VAL	2DEC	1.5569 B-10	# FAPS (3500 LBS THRUST)

S40.136	2DEC	.4671 B-9	# .4671 M NEWTONS (DPS)
S40.136_	2DEC	.4671 B+1	# S40.136 SHIFTED LEFT 10.

SETLOC ASENT1
BANK
COUNT* \$\$/P70

(1/DV)A	2DEC	15.20 B-7	# 2 SECONDS WORTH OF INITIAL ASCENT
---------	------	-----------	-------------------------------------


```
1
2                                     # STAGE ACCELERATION -- INVERTED (M/CS)
3                                     # 1) PREDICATED ON A LIFTOFF MASS OF
4                                     #   4869.9 KG (SNA-8-D-027 7/11/68)
5                                     # 2) PREDICATED ON A CONTRIBUTION TO VEH-
6                                     #   ICLE ACCELERATION FROM RCS THRUSTERS
7                                     #   EQUIV. TO 1 JET ON CONTINUOUSLY.
8
9      K(1/DV)          2DEC      436.70 B-9      # DPS ENGINE THRUST IN NEWTONS / 100 CS.
10
11      (AT)A           2DEC      3.2883 E-4 B9    # INITIAL ASC. STG. ACCELERATION ** M/CS.
12                                     # ASSUMPTIONS SAME AS FOR (1/DV)A.
13      (TBUP)A         2DEC      91902 B-17      # ESTIMATED BURN-UP TIME OF THE ASCENT STG.
14                                     # ASSUMPTIONS SAME AS FOR (1/DV)A WITH THE
15                                     # ADDITIONAL ASSUMPTION THAT NET MASS-FLOW
16                                     # RATE = 5.299 KG/SEC = 5.135 (APS) +
17                                     # .164 (1 RCS JET).
18
19      SETLOC  ASENT
20      BANK
21      COUNT*  $$/ASENT
22      AT/RCS  2DEC      .0000785 B+10          # 4 JETS IN A DRY LEM
23
24      SETLOC  SERVICES
25      BANK
26      COUNT*  $$/SERV
27
28      # *** THE ORDER OF THE FOLLOWING TWO CONSTANTS MUST NOT BE CHANGED *****
29
30      APSVEX      DEC      -3030 E-2 B-5      # 9942 FT/SEC IN M/CS.
31      DPSVEX      DEC*     -2.95588868 E+1 B-05*  # VE (DPS) +2.95588868E+ 3
32
33      # *****
34
35      SETLOC  F2DPS*31
36      BANK
37      COUNT*  $$/F2DPS
38
39      TRIMACCL     2DEC*     +3.50132708 E-5 B+08*  # A (T) +3.50132708E- 1
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

CONTROLLED_CONSTANTS

THROTTLING AND THRUST DETECTION PARAMETERS

SETLOC P40S
BANK
COUNT* \$\$/P40

THRESH1 DEC 24
THRESH3 DEC 12
HIRTHROT = BIT13

SETLOC FFTAG5
BANK
COUNT* \$\$/P40

THRESH2 DEC 308

SETLOC FTHROT
BANK
COUNT* \$\$/THROT

FMAXODD DEC +3841 # FSAT +4.81454413 E+4
FMAXPOS DEC +3467 # FMAX +4.34546769 E+4
THROTLAG DEC +20 # TAU (TH) +1.99999999 E-1
SCALEFAC 2DEC* +7.97959872 E+2 B-16* # BITPERF +7.97959872 E-2

SETLOC F2DPS*32
BANK
COUNT* \$\$/F2DPS

DPSTHRSH DEC 36 # (THRESH1 + THRESH3 FOR P63)

LM HARDWARE-RELATED PARAMETERS

SETLOC RADARUPT
BANK
COUNT* \$\$/RRUPT

LVELBIAS DEC -12288 # LANDING RADAR BIAS FOR 153.6 KC.
RDOTBIAS 2DEC 17000 # BIAS COUNT FOR RR RANGE RATE.

SETLOC LRS22
BANK
COUNT* \$\$/LRS22

RDOTCONV 2DEC -.0019135344 B7 # CONVERTS RR RDOT READING TO M/CS AT 2(7)
RANGCONV 2DEC 2.859024 B-3 # CONVERTS RR RANGE READING TO M. AT 2(-29

SETLOC SERVICES
BANK
COUNT* \$\$/SERV

HBEAMANT 2DEC -.4687018041 # RANGE BEAM IN LR ANTENNA COORDINATES.
2DEC 0
2DEC -.1741224271

HSCAL 2DEC -.3288792 # SCALES 1.079 FT/BIT TO 2(22)M.

***** THE SEQUENCE OF THE FOLLOWING CONSTANTS MUST BE PRESERVED *****

VZSCAL 2DEC +.5410829105 # SCALES .8668 FT/SEC/BIT TO 2(18) M/CS.
VYSCAL 2DEC +.7565672446 # SCALES 1.212 FT/SEC/BIT TO 2(18) M/CS.
VXSCAL 2DEC -.4020043770 # SCALES -.644 FT/SEC/BIT TO 2(18) M/CS.

KPIP DEC .0512 # SCALES DELV TO UNITS OF 2(5) M/CS.
KPIP1 2DEC .0128 # SCALES DELV TO UNITS OF 2(7) M/CS.
KPIP2 2DEC .0064 # SCALES DELV TO UNITS OF 2(8) M/CS.

CONTROLLED_CONSTANTS

1						1
2	ALTCONV	2DEC	1.399078846 B-4	# CONVERTS M*2(-24) TO BIT UNITS *2(-28).		2
3	ARCONV1	2DEC	656.167979 B-10	# CONV. ALTRATE COMP. TO BIT UNITS<		3
4						4
5		SETLOC	R10			5
6		BANK				6
7		COUNT*	\$\$/R10			7
8						8
9	ARCONV	OCT	24402	# 656.1679798B-10 CONV ALTRATE TO BIT UNIT		9
10	ARTOA	DEC	.1066098 B-1	# .25/2.345 B-1 4X/SEC CYCLE RATE.		10
11	ARTOA2	DEC	.0021322 B8	# (.5)/(2.345)(100)		11
12	VELCONV	OCT	22316	# 588.914 B-10 CONV VEL. TO BIT UNITS.		12
13	KPIP1(5)	DEC	.0512	# SCALES DELV TO M/CS*2(-5).		13
14	MAXVBITS	OCT	00547	# MAX. DISPLAYED VELOCITY 199.9989 FT/SEC.		14
15						15
16		SETLOC	DAPS3			16
17		BANK				17
18		COUNT*	\$\$/DAPAD			18
19						19
20	TORKJET1	DEC	.03757	# 550 / .2 SCALED AT (+16) 64 / 180		20
21						21
22						22
23						23
24						24
25						25
26						26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

CONTROLLED_CONSTANTS

PARAMETERS RELATING TO MASS, INERTIA, AND VEHICLE DIMENSIONS

SETLOC FRANDRES
BANK
COUNT* \$\$/START

FULLAPS DEC 5050 B-16 # NOMINAL FULL ASCENT MASS -- 2(16) KG.

SETLOC LOADDAP1
BANK
COUNT* \$\$/R03

MINLMD DEC -2850 B-16 # MIN. DESCENT STAGE MASS -- 2(16) KG.
MINMINLM DEC -2200 B-16 # MIN ASCENT STAGE MASS -- 2(16) KG.
MINCSM = BIT11 # MIN CSM MASS (OK FOR 1/ACCS) = 9050 LB

SETLOC DAPS3
BANK
COUNT* \$\$/DAPAD

LOASCENT DEC 2200 B-16 # MIN ASCENT LEM MASS -- 2(16) KG.
HIDESCNT DEC 15300 B-16 # MAX DESCENT LEM MASS -- 2(16) KG.
LODESCNT DEC 1750 B-16 # MIN DESCENT STAGE (ALONE) -- 2(16) KG.

1412THE
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
1

CONTROLLED_CONSTANTS

PHYSICAL CONSTANTS (TIME - INVARIANT)

SETLOC IMU2
BANK
COUNT* \$\$/P07

OMEG/MS 2DEC .24339048

SETLOC R30LOC
BANK
COUNT* \$\$/R30

*** THE ORDER OF THE FOLLOWING TWO CONSTANTS MUST BE PRESERVED *****

1/RTMUM 2DEC* .45162595 E-4 B14*
1/RTMUE 2DEC* .50087529 E-5 B17*

SETLOC P40S1
BANK
COUNT* \$\$/S40.9

EARTHMU 2DEC* -3.986032 E10 B-36* # M(3)/CS(2)

SETLOC ASENT1
BANK
COUNT* \$\$/P12

MUM(-37) 2DEC* 4.9027780 E8 B-37*
MOONRATE 2DEC* .26616994890062991 E-7 B+19* # RAD/CS.

SETLOC SERVICES
BANK
COUNT* \$\$/SERV

*** THE ORDER OF THE FOLLOWING TWO CONSTANTS MUST BE PRESERVED *****

-MUDT 2DEC* -7.9720645 E+12 B-44*
-MUDT1 2DEC* -9.8055560 E+10 B-44*

-MUDTMUN 2DEC* -9.8055560 E+10 B-38*
RESQ 2DEC* 40.6809913 E12 B-58*

1412THE

CONTROLLED_CONSTANTS

PHYSICAL CONSTANTS (TIME - VARIANT)

SETLOC STARTAB
BANK
COUNT* \$\$/STARS

2DEC	+.8342971408 B-1	# STAR 37	X
2DEC	-.2392481515 B-1	# STAR 37	Y
2DEC	-.4966976975 B-1	# STAR 37	Z
2DEC	+.8139832631 B-1	# STAR 36	X
2DEC	-.5557243189 B-1	# STAR 36	Y
2DEC	+.1691204557 B-1	# STAR 36	Z
2DEC	+.4541086270 B-1	# STAR 35	X
2DEC	-.5392368197 B-1	# STAR 35	Y
2DEC	+.7092312789 B-1	# STAR 35	Z
2DEC	+.3201817378 B-1	# STAR 34	X
2DEC	-.4436021946 B-1	# STAR 34	Y
2DEC	-.8370786986 B-1	# STAR 34	Z
2DEC	+.5520184464 B-1	# STAR 33	X
2DEC	-.7933187400 B-1	# STAR 33	Y
2DEC	-.2567508745 B-1	# STAR 33	Z
2DEC	+.4537196908 B-1	# STAR 32	X
2DEC	-.8779508801 B-1	# STAR 32	Y
2DEC	+.1527766153 B-1	# STAR 32	Z
2DEC	+.2069525789 B-1	# STAR 31	X
2DEC	-.8719885748 B-1	# STAR 31	Y
2DEC	-.4436288486 B-1	# STAR 31	Z
2DEC	+.1217293692 B-1	# STAR 30	X
2DEC	-.7702732847 B-1	# STAR 30	Y

1

# CONTROLLED CONSTANTS							PAGE 17
2	2DEC	-.4523440203	B-1	# STAR 21	X		
3	2DEC	-.0493710140	B-1	# STAR 21	Y		
4	2DEC	-.8904759346	B-1	# STAR 21	Z		
5							
6	2DEC	-.9525211695	B-1	# STAR 20	X		
7	2DEC	-.0593434796	B-1	# STAR 20	Y		
8	2DEC	-.2986331746	B-1	# STAR 20	Z		
9							
10	2DEC	-.9656605484	B-1	# STAR 19	X		
11	2DEC	+.0525933156	B-1	# STAR 19	Y		
12	2DEC	+.2544280809	B-1	# STAR 19	Z		
13							
14	2DEC	-.8608205219	B-1	# STAR 18	X		
15	2DEC	+.4636213989	B-1	# STAR 18	Y		
16	2DEC	+.2098647835	B-1	# STAR 18	Z		
17							
18	2DEC	-.7742591356	B-1	# STAR 17	X		
19	2DEC	+.6152504197	B-1	# STAR 17	Y		
20	2DEC	-.1482892839	B-1	# STAR 17	Z		
21							
22	2DEC	-.4657947941	B-1	# STAR 16	X		
23	2DEC	+.4774785033	B-1	# STAR 16	Y		
24	2DEC	+.7450164351	B-1	# STAR 16	Z		
25							
26	2DEC	-.3612508532	B-1	# STAR 15	X		
27	2DEC	+.5747270840	B-1	# STAR 15	Y		
28	2DEC	-.7342932655	B-1	# STAR 15	Z		
29							
30	2DEC	-.4118589524	B-1	# STAR 14	X		
31	2DEC	+.9065485360	B-1	# STAR 14	Y		
32	2DEC	+.0924226975	B-1	# STAR 14	Z		
33							
34	2DEC	-.1820751783	B-1	# STAR 13	X		
35							
36							
37							
38							
39							
40							
41							
42							
43							
44							
45							
46							
47							
48							
49							
50							
51							
52							
53							
54							
55							
56							
57							
58							
59							
60							

# CONTROLLED CONSTANTS						PAGE 50
2	2DEC	+.9404899869	B-1	# STAR 13	Y	
3	2DEC	-.2869271926	B-1	# STAR 13	Z	
5	2DEC	-.0614937230	B-1	# STAR 12	X	
6	2DEC	+.6031563286	B-1	# STAR 12	Y	
7	2DEC	-.7952489957	B-1	# STAR 12	Z	
9	2DEC	+.1371725575	B-1	# STAR 11	X	
10	2DEC	+.6813721061	B-1	# STAR 11	Y	
11	2DEC	+.7189685267	B-1	# STAR 11	Z	
13	2DEC	+.2011399589	B-1	# STAR 10	X	
14	2DEC	+.9690337941	B-1	# STAR 10	Y	
15	2DEC	-.1432348512	B-1	# STAR 10	Z	
17	2DEC	+.3507315038	B-1	# STAR 9	X	
18	2DEC	+.8926333307	B-1	# STAR 9	Y	
19	2DEC	+.2831839492	B-1	# STAR 9	Z	
21	2DEC	+.4105636020	B-1	# STAR 8	X	
22	2DEC	+.4988110001	B-1	# STAR 8	Y	
23	2DEC	+.7632988371	B-1	# STAR 8	Z	
25	2DEC	+.7032235469	B-1	# STAR 7	X	
26	2DEC	+.7075846047	B-1	# STAR 7	Y	
27	2DEC	+.0692868685	B-1	# STAR 7	Z	
29	2DEC	+.5450107404	B-1	# STAR 6	X	
30	2DEC	+.5314955466	B-1	# STAR 6	Y	
31	2DEC	-.6484410356	B-1	# STAR 6	Z	
33	2DEC	+.0130968840	B-1	# STAR 5	X	
34	2DEC	+.0078062795	B-1	# STAR 5	Y	
37						
38						
39						
40						
41						
42						
43						
44						
45						
46						
47						
48						
49						
50						
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						

1						1
2						2
3						3
4						4
5						5
6						6
7						7
8						8
9						9
10						10
11						11
12						12
13						13
14						14
15						15
16						16
17						17
18						18
19						19
20						20
21						21
22						22
23						23
24						24
25						25
26						26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1									1
2		2DEC	.082354 B-1	# K4 COS(OBL)SIN(IM) *					2
3	CSTODAY	2DEC	8640000 B-33	#	* NOTE:	*			3
4	RCB-13	OCT	00002	#	* TABLES CONTAIN	*			4
5		OCT	00000	#	* CONSTANTS FOR	*			5
6	RATESP	2DEC	.03660098 B+4	# LOMR	* 1969 - 1970	*			6
7		2DEC	.00273779 B+4	# LOSR					7
8		2DEC	-.00014719 B+4	# LONR					8
9		2DEC	.815282336	# LOMO					9
10		2DEC	.274674910	# LOSO					10
11		2DEC	.986209499	# LONO					11
12	VAL67	2DEC*	.01726666666 B+1*	# AMOD					12
13		2DEC	.530784445	# AARG					13
14		2DEC	.036291712 B+1	# 1/27					14
15		2DEC	.003505277 B+1	# BMOD					15
16		2DEC	.585365625	# BARG					16
17		2DEC	.03125 B+1	# 1/32					17
18		2DEC	.005325277 B+1	# CMOD					18
19		2DEC	-.01106341036	# CARG					19
20		2DEC	.002737925 B+1	# 1/365					20
21									21
22	# *****								22
23									23
24		SETLOC	PLANTIN2						24
25		BANK							25
26		COUNT*	\$\$/LUR0T						26
27									27
28	COSI	2DEC	.99964173 B-1	# COS (5521.5 SEC.) B-1					28
29	SINI	2DEC	.02676579 B-1	# SIN (5521.5 SEC.) B-1					29
30	NODDOT	2DEC	-.457335121 E-2	# REV/CSEC B+28 = -1.07047011 E-8 RAD/SEC					30
31	FDOT	2DEC	.570863327	# REV/CSEC B+27 = 2.67240410 E-6 RAD/SEC					31
32									32
33									33
34									34
35									35
36									36
37									37
38									38
39									39
40									40
41									41
42									42
43									43
44									44
45									45
46									46
47									47
48									48
49									49
50									50
51									51
52									52
53									53
54									54
55									55
56									56
57									57
58									58
59									59
60									60

PAGE 53

#	REV/CSEC	B+28	=	-7.19757301	E-14	RAD/SEC
#	REVS	B-D	=	6.19653663041		RAD
#	REVS	B-D	=	5.20932947829		RAD
#	REVS	B-D	=	0.40916190299		RAD
#	REV/CSEC	B+23	=	7.29211494	E-5	RAD/SEC

*** CHANNEL DESCRIPTIONSF WORDS ARE ALLOCATED IN ERASABLE ASSIGNMENTS ***

CHANNEL 1 IDENTICAL TO COMPUTER REGISTER L (0001)

CHANNEL 2 IDENTICAL TO COMPUTER REGISTER Q (0002)

CHANNEL 3 HISCALAR; INPUT CHANNEL; MOST SIGNIFICANT 14 BITS FROM 33 STAGE BINARY COUNTER. SCALE FACTOR IS B23 IN CSEC, SO MAX VALUE ABOUT 23.3 HOURS AND LEAST SIGNIFICANT BIT 5.12 SECS.

CHANNEL 4 LOSCALAR; INPUT CHANNEL; NEXT MOST SIGNIFICANT 14 BITS FROM THE 33 STAGE BINARY COUNTER ASSOCIATED WITH CHANNEL 3. SCALE FACTOR IS B9 IN CSEC. SO MAX VAL IS 5.12 SEC AND LEAST SIGNIFICANT BIT IS 1/3200 SEC. SCALE FACTOR OF D.P. WORD WITH CHANNEL 3 IS B23 CSEC.

CHANNEL 5 PYJETS; OUTPUT CHANNEL; PITCH RCS JET CONTROL. (REACTION CONTROL SYSTEM) USES BITS 1-8.

CHANNEL 6 ROLLJETS; OUTPUT CHANNEL; ROLL RCS JET CONTROL. (REACTION CONTROL SYSTEM) USES BIT 1-8.

CHANNEL 7 SUPERBNK; OUTPUT CHANNEL; NOT RESET BY RESTART; FIXED EXTENSION BITS USED TO SELECT THE APPROPRIATE FIXED MEMORY BANK IF FBANK IS 30 OCTAL OR MORE. USES BITS 5-7.

CHANNEL 10 OUT0; OUTPUT CHANNEL; REGISTER USED TO TRANSMIT LATCHING-RELAY DRIVING INFORMATION FOR THE DISPLAY SYSTEM. BITS 15-12 ARE SET TO THE ROW NUMBER (1-14 OCTAL) OF THE RELAY TO BE CHANGED AND BITS 11-1 CONTAIN THE REQUIRED SETTINGS FOR THE RELAYS IN THE ROW.

CHANNEL 11 DSALMOUT; OUTPUT CHANNEL; REGISTER WHOSE BITS ARE USED FOR ENGINE ON-OFF CONTROL AND TO DRIVE INDIVIDUAL INDICATORS OF THE DISPLAY SYSTEM. BITS 1-7 ARE A RELAYS.

BIT 1 ISS WARNING

BIT 2 LIGHT COMPUTER ACTIVITY LAMP

BIT 3 LIGHT UPLINK ACTIVITY LAMP

BIT 4 LIGHT TEMP CAUTION LAMP

BIT 5 LIGHT KEYBOARD RELEASE LAMP

BIT 6 FLASH VERB AND NOUN LAMPS

BIT 7 LIGHT OPERATOR ERROR LAMP

1412THE

INPUT_OUTPUT_CHANNEL_BIT_DESCRIPTIONS

1					1
2	#	CHANNEL 13	CHAN13; OUTPUT CHANNEL.		2
3	#				3
4	#	BIT 1	RADAR C	PROPER SETTING OF THE A,B,C MATRIX	4
5	#	BIT 2	RADAR B	SELECTS CERTAIN RADAR	5
6	#	BIT 3	RADAR A	PARAMETERS TO BE READ.	6
7	#	BIT 4	RADAR ACTIVITY		7
8	#	BIT 5	NOT USED (CONNECTS AN ALTERNATE INPUT TO UPLINK)		8
9	#	BIT 6	BLOCK INPUTS TO UPLINK CELL		9
10	#	BIT 7	DOWNLINK TELEMETRY WORD ORDER CODE BIT		10
11	#	BIT 8	RHC COUNTER ENABLE (READ HAND CONTROLLER ANGLES)		11
12	#	BIT 9	START RHC READ INTO COUNTERS IF BIT 8 SET		12
13	#	BIT 10	TEST ALARMS, TEST DSKY LIGHTS		13
14	#	BIT 11	ENABLE STANDBY		14
15	#	BIT 12	RESET TRAP 31-A	ALWAYS APPEAR TO BE SET TO 0	15
16	#	BIT 13	RESET TRAP 31-B	ALWAYS APPEAR TO BE SET TO 0	16
17	#	BIT 14	RESET TRAP 32	ALWAYS APPEAR TO BE SET TO 0	17
18	#	BIT 15	ENABLE T6 RUPT		18
19					19
20	#	CHANNEL 14	CHAN14; OUTPUT CHANNEL; USED TO CONTROL COMPUTER COUNTER CELLS (CDU,GYRO,SPACECRAFT FUNC.		20
21	#				21
22	#	BIT 1	OUTLINK ACTIVITY (NOT USED)		22
23	#	BIT 2	ALTITUDE RATE OR ALTITUDE SELECTOR		23
24	#	BIT 3	ALTITUDE METER ACTIVITY		24
25	#	BIT 4	THRUST DRIVE ACTIVITY FOR DESCENT ENGINE		25
26	#	BIT 5	SPARE		26
27	#	BIT 6	GYRO ENABLE POWER FOR PULSES		27
28	#	BIT 7	GYRO SELECT B	PAIR OF BITS IDENTIFIES AXIS OF -	28
29	#	BIT 8	GYRO SELECT A	GYRO SYSTEM TO BE TORQUED.	29
30	#	BIT 9	GYRO TORQUING COMMAND IN NEGATIVE DIRECTION		30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1412THE

1

INPUT_OUTPUT_CHANNEL_BIT_DESCRIPTIONS

#	BIT 10	GYRO ACTIVITY
#	BIT 11	DRIVE CDU S
#	BIT 12	DRIVE CDU T
#	BIT 13	DRIVE CDU Z
#	BIT 14	DRIVE CDU Y
#	BIT 15	DRIVE CDU X
#	CHANNEL 15	MNKEYIN; INPUT CHANNEL; KEY CODE INPUT FROM KEYBOARD OF DSKY, SENSED BY PROGRAM WHEN PROGRAM INTERRUPT #5 IS RECEIVED. USES BITS 5-1
#	CHANNEL 16	NAVKEYIN; INPUT CHANNEL; OPTICS MARK INFORMATION AND NAVIGATION PANEL DSKY (CM) OR THRUST CONTROL (LM) SENSED BY PROGRAM WHEN PROGRAM INTERRUPT #6 IS RECEIVED. USES BITS 3-7 ONLY.
#	BIT 1	NOT ASSIGNED
#	BIT 2	NOT ASSIGNED
#	BIT 3	OPTICS X-AXIS MARK SIGNAL FOR ALIGN OPTICAL TSCOPE
#	BIT 4	OPTICS Y-AXIS MARK SIGNAL FOR AOT
#	BIT 5	OPTICS MARK REJECT SIGNAL
#	BIT 6	DESCENT+ ; CREW DESIRED SLOWING RATE OF DESCENT
#	BIT 7	DESCENT- ; CREW DESIRED SPEEDING UP RATE OF DESCENT
# NOTE: ALL BITS IN CHANNELS 30-33 ARE INVERTED AS SENSED BY THE PROGRAM, SO THAT A VALUE OF ZERO MEANS THAT THE INDICATED SIGNAL IS PRESENT.		
#	CHANNEL 30	INPUT CHANNEL
#	BIT 1	ABORT WITH DESCENT STAGE
#	BIT 2	UNUSED
#	BIT 3	ENGINE ARMED SIGNAL
#	BIT 4	ABORT WITH ASCENT ENGINE STAGE
#	BIT 5	AUTO THROTTLE; COMPUTER CONTROL OF DESCENT ENGINE

#	INPUT	OUTPUT	CHANNEL	BIT	DESCRIPTIONS
1					
2	#			BIT 6	DISPLAY INERTIAL DATA
3	#			BIT 7	RR CDU FAIL
4	#			BIT 8	SPARE
5	#			BIT 9	IMU OPERATE WITH NO MALFUNCTION
6	#			BIT 10	LM COMPUTER (NOT AGS) HAS CONTROL OF LM
7	#			BIT 11	IMU CAGE COMMAND TO DRIVE IMU GIMBAL ANGLES TO 0.
8	#			BIT 12	IMU CDU FAIL (MALFUNCTION OF IMU CDU,S)
9	#			BIT 13	IMU FAIL (MALFUNCTION OF IMU STABILIZATION LOOPS)
10	#			BIT 14	ISS TURN ON REQUESTED
11	#			BIT 15	TEMPERATURE OF STABLE MEMBER WITHIN DESIGN LIMITS
12					
13	#		CHANNEL 31		INPUT CHANNEL; BITS ASSOCIATED WITH THE ATTITUDE CONTROLLER, TRANSLATIONAL CONTROLLER,
14	#				AND SPACECRAFT ATTITUDE CONTROL; USED BY RCS DAP
15	#				
16	#			BIT 1	ROTATION (BY RHC) COMMANDED IN POSITIVE PITCH DIRECTION; MUST BE IN MINIMUM IMPULSE MODE.
17	#				ALSO POSITIVE ELEVATION CHANGE FOR LANDING POINT DESIGNATOR
18	#			BIT 2	AS BIT 1 EXCEPT NEGATIVE PITCH AND ELEVATION
19	#			BIT 3	ROTATION (BY RHC) COMMANDED IN POSITIVE YAW DIRECTION; MUST BE IN MINIMUM IMPULSE MODE.
20	#			BIT 4	AS BIT 3 EXCEPT NEGATIVE YAW
21	#			BIT 5	ROTATION (BY RHC) COMMANDED IN POSITIVE ROLL DIRECTION; MUST BE IN MINIMUM IMPULSE MODE.
22	#				ALSO POSITIVE AZIMUTH CHANGE FOR LANDING POINT DESIGNATOR
23	#			BIT 6	AS BIT 5 EXCEPT NEGATIVE ROLL AND AZIMUTH
24	#			BIT 7	TRANSLATION IN +X DIRECTION COMMANDED BY THC
25	#			BIT 8	TRANSLATION IN -X DIRECTION COMMANDED BY THC
26	#			BIT 9	TRANSLATION IN +Y DIRECTION COMMANDED BY THC
27	#			BIT 10	TRANSLATION IN -Y DIRECTION COMMANDED BY THC
28	#			BIT 11	TRANSLATION IN +Z DIRECTION COMMANDED BY THC
29	#			BIT 12	TRANSLATION IN -Z DIRECTION COMMANDED BY THC
30					
31					
32					
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46					
47					
48					
49					
50					
51					
52					
53					
54					
55					
56					
57					
58					
59					
60					

INPUT_OUTPUT_CHANNEL_BIT_DESCRIPTIONS

1				1
2	#	BIT 13	ATTITUDE HOLD MODE ON SCS MODE CONTROL SWITCH	2
3	#	BIT 14	AUTO STABILIZATION OF ATTITUDE ON SCS MODE SWITCH	3
4	#	BIT 15	ATTITUDE CONTROL OUT OF DETENT (RHC NOT IN NEUTRAL	4
5				5
6	#	CHANNEL 32	INPUT CHANNEL.	6
7	#			7
8	#	BIT 1	THRUSTERS 2 & 4 DISABLED BY CREW	8
9	#	BIT 2	THRUSTERS 5 & 8 DISABLED BY CREW	9
10	#	BIT 3	THRUSTERS 1 & 3 DISABLED BY CREW	10
11	#	BIT 4	THRUSTERS 6 & 7 DISABLED BY CREW	11
12	#	BIT 5	THRUSTERS 14 & 16 DISABLED BY CREW	12
13	#	BIT 6	THRUSTERS 13 & 15 DISABLED BY CREW	13
14	#	BIT 7	THRUSTERS 9 & 12 DISABLED BY CREW	14
15	#	BIT 8	THRUSTERS 10 & 11 DISABLED BY CREW	15
16	#	BIT 9	DESCENT ENGINE GIMBALS DISABLED BY CREW	16
17	#	BIT 10	APPARENT DESCENT ENGINE GIMBAL FAILURE	17
18	#	BIT 14	INDICATES PROCEED KEY IS DEPRESSED	18
19				19
20	#	CHANNEL 33	CHAN33; INPUT CHANNEL; FOR HARDWARE STATUS AND COMMAND INFORMATION. BITS 15-11 ARE FLIP-	20
21	#		FLOP BITS RESET BY A CHANNEL "WRITE" COMMAND THAT ARE RESET BY A RESTART & BY T4RUPT LOOP.	21
22	#			22
23	#	BIT 1	SPARE	23
24	#	BIT 2	RR AUTO-POWER ON	24
25	#	BIT 3	RR RANGE LOW SCALE	25
26	#	BIT 4	RR DATA GOOD	26
27	#	BIT 5	LR RANGE DATA GOOD	27
28	#	BIT 6	LR POS1	28
29	#	BIT 7	LR POS2	29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	# INPUT COUNT OF CHANNEL BIT DESCRIPTION			1
2	#	BIT 8	LR VEL DATA GOOD	2
3	#	BIT 9	LR RANGE LOW SCALE	3
4	#	BIT 10	BLOCK UPLINK INPUT	4
5	#	BIT 11	UPLINK TOO FAST	5
6	#	BIT 12	DOWNLINK TOO FAST	6
7	#	BIT 13	PIPA FAIL	7
8	#	BIT 14	WARNING OF REPEATED ALARMS: RESTART,COUNTER FAIL, VOLTAGE FAIL,AND SCALAR DOUBLE.	8
9	#	BIT 15	LGC OSCILLATOR STOPPED	9
10				10
11	#	CHANNEL 34	DNT M1; OUTPUT CHANNEL; DOWNLINK 1 FIRST OF TWO WORDS SERIALIZATION.	11
12	#	CHANNEL 35	DNT M2; OUTPUT CHANNEL DOWNLINK 2 SOCOND OF TWO WORDS SERIALIZATION.	12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

FLAGWORDS 0-11 ARE DOWNLINKED AND CAN BE SET AND CLEARED BY UP-FLAG AND DOWN-FLAG INSTRUCTIONS IN THE INTERPRETER. THESE WERE PREVIOUSLY LISTED UNDER "INTERPRETIVE SWITCH BIT ASSIGNMENTS" IN THE ERASABLE LOG SECTION. FLAGWORDS 12 & 13 WERE PREVIOUSLY RADMODES AND DAPBOOLS AND ARE STILL DOWNLINKED UNDER THOSE NAMES.

ALPHABETICAL LIST OF FLAGWORDS

# FLAGWORD	DEC. NUMBER	BIT AND FLAG	BIT NAME
# ACCOKFLG	207	BIT 3 FLAG 13	ACCSOKAY
# ACC4-2FL	199	BIT 11 FLAG 13	ACC4OR2X
# ACMODFLG	032	BIT 13 FLAG 2	ACMODBIT
# ALTSCALE	186	BIT 9 FLAG 12	ALTSCBIT
# ANTENFLG	183	BIT 12 FLAG 12	ANTENBIT
# AORBSFLG	205	BIT 5 FLAG 13	AORBSYST
# AORBTFLG	200	BIT 10 FLAG 13	AORBTRAN
# APSESW	130	BIT 5 FLAG 8	APSESBIT
# APSFLAG	152	BIT 13 FLAG 10	APSFLBIT
# ASTNFLAG	108	BIT 12 FLAG 7	ASTNBIT
# ATTFLAG	104	BIT 1 FLAG 6	ATTFLBIT
# AUTOMODE	193	BIT 2 FLAG 12	AUTOMBIT
# AUTR1FLG	209	BIT 1 FLAG 13	AUTRATE1
# AUTR2FLG	208	BIT 2 FLAG 13	AUTRATE2
# AUXFLAG	103	BIT 2 FLAG 6	AUXFLBIT
# AVEGFLAG	115	BIT 5 FLAG 7	AVEGFBIT
# AVEMIDSW	149	BIT 1 FLAG 9	AVEMDBIT
# AVFLAG	040	BIT 5 FLAG 2	AVFLBIT
# CALCMAN2	043	BIT 2 FLAG 2	CALC2BIT
# CALCMAN3	042	BIT 3 FLAG 2	CALC3BIT
# CDESFLAG	180	BIT 15 FLAG 12	CDESBIT
# CMOONFLG	123	BIT 12 FLAG 8	CMOONBIT
# COGAFLAG	131	BIT 4 FLAG 8	COGAFBIT
# CSMDKFLG	197	BIT 13 FLAG 13	CSMDOCKD
# CULTFLAG	053	BIT 7 FLAG 3	CULTBIT
# DAPBOOLS		FLGWRD13	
# DBSELFLG	206	BIT 4 FLAG 13	DBSELECT
# DESIGFLG	185	BIT 10 FLAG 12	DESIGBIT
# DIDFLAG	016	BIT 14 FLAG	DIDFLBIT
# DIMOFLAG	059	BIT 1 FLAG 3	DIMOBIT
# DMENFLG	081	BIT 9 FLAG 5	DMENFBIT
# DRIFTDFL	202	BIT 8 FLAG 13	DRIFTBIT
# DRIFTFLG	030	BIT 15 FLAG 2	DRFTBIT
# DSKYFLAG	075	BIT 15 FLAG 5	DSKYFBIT

# FLAGWORD ASSIGNMENTS					PAGE 02	
1						
2	# D6OR9FLG	058	BIT 2	FLAG 3	D6OR9BIT	
3	# ENGONFLG	083	BIT 7	FLAG 5	ENGONBIT	
4	# ERADFLAG	017	BIT 13	FLAG 1	ERADFBIT	
5	# ETPIFLAG	038	BIT 7	FLAG 2	ETPIBIT	EQUIVALENT FLAG NAME: DPTNSW
6	# FINALFLG	039	BIT 6	FLAG 2	FINALBIT	
7	# FLAGWRD0	(000-014)	(STATE +0)			
8	# FLAGWRD1	(015-029)	(STATE +1)			
9	# FLAGWRD2	(030-044)	(STATE +2)			
10	# FLAGWRD3	(045-059)	(STATE +3)			
11	# FLAGWRD4	(060-074)	(STATE +4)			
12	# FLAGWRD5	(075-089)	(STATE +5)			
13	# FLAGWRD6	(090-104)	(STATE +6)			
14	# FLAGWRD7	(105-119)	(STATE +7)			
15	# FLAGWRD8	(120-134)	(STATE +8D)			
16	# FLAGWRD9	(135-149)	(STATE +9D)			
17	# FLAP	142	BIT 8	FLAG 9	FLAPBIT	
18	# FLGWRD10	(150-164)	(STATE +10D)			
19	# FLGWRD11	(165-179)	(STATE +11D)			
20	# FLGWRD12	(180-194)	(STATE +12D)			
21	# FLGWRD13	(195-209)	(STATE +13D)			
22	# FLPC	138	BIT 12	FLAG 9	FLPCBIT	
23	# FLPI	139	BIT 11	FLAG 9	FLPIBIT	
24	# FLRCS	149	BIT 10	FLAG 9	FLRCSBIT	
25	# FLUNDISP	125	BIT 10	FLAG 8	FLUNDBIT	
26	# FLVR	136	BIT 14	FLAG 9	FLVRBIT	
27	# FREEFLAG	012	BIT 3	FLAG 0	FREEFBIT	
28	# FSPASFLG	005	BIT 10	FLAG 0	FSPASBIT	
29	# GLOKFAIL	046	BIT 14	FLAG 3	GLOKFBIT	
30	# GMBDRVSW	095	BIT 10	FLAG 6	GMBDRBIT	
31	# GUESSW	028	BIT 2	FLAG 1	GUESSBIT	
32	# HFLSHFLG	179	BIT 1	FLAG 11	HFLSHBIT	
33	# IDLEFLAG	113	BIT 7	FLAG 7	IDLEFBIT	
34	# IGNFLAG	107	BIT 13	FLAG 7	IGNFLBIT	
35	# IMPULSW	036	BIT 9	FLAG 2	IMPULBIT	
36	# IMUSE	007	BIT 8	FLAG 0	IMUSEBIT	
37	# INFINFLG	128	BIT 7	FLAG 8	INFINBIT	
38	# INITALGN	133	BIT 2	FLAG 8	INITABIT	
39	# INTFLAG	151	BIT 14	FLAG 10	INTFLBIT	
40	# INTYPFLG	056	BIT 4	FLAG 3	INTYPBIT	
41	# ITSWICH	105	BIT 15	FLAG 7	ITSWBIT	
42	# JSWITCH	001	BIT 14	FLAG 0	JSWCHBIT	
43	# LETABORT	141	BIT 9	FLAG 9	LETABBIT	
44	# LMOONFLG	124	BIT 11	FLAG 8	LMOONBIT	
45	# LOKONSW	010	BIT 5	FLAG 0	LOKONBIT	
46	# LOSCMFLG	033	BIT 12	FLAG 2	LOSCMBIT	
47	# LRALTFLG	190	BIT 5	FLAG 12	LRALTBIT	
48	# LRBYPASS	165	BIT 15	FLAG 11	LRBYBIT	
49	# LRINH	172	BIT 8	FLAG 11	LRINHBIT	
50	# LRPOSFLG	189	BIT 6	FLAG 12	LRPOSBIT	
51	# LRVELFLG	187	BIT 8	FLAG 12	LRVELBIT	
52	# PAGE63					
53	# LUNAFLAG	048	BIT 12	FLAG 3	LUNABIT	
54	# MANUFLAG	106	BIT 14	FLAG 7	MANUFBIT	
55	# MGLVFLAG	088	BIT 2	FLAG 5	MGLVFBIT	
56	# MIDAVFLG	148	BIT 2	FLAG 9	MIDAVBIT	
57	# MIDFLAG	002	BIT 13	FLAG 0	MIDFLBIT	
58	# MID1FLAG	147	BIT 3	FLAG 9	MID1BIT	
59	# MKOVFLAG	072	BIT 3	FLAG 4	MKOVBIT	
60	# MOONFLAG	003	BIT 12	FLAG 0	MOONBIT	

	#	MRKIDFLG	060	BIT 15	FLAG 4	MRKIDBIT
1	#	MRKNVFLG	066	BIT 9	FLAG 4	MRKNVBIT
2	#	MRUPTFLG	070	BIT 5	FLAG 4	MRUPTBIT
3	#	MUNFLAG	097	BIT 8	FLAG 6	MUNFLBIT
4	#	MWAITFLG	064	BIT 11	FLAG 4	MWAITBIT
5	#	NEEDLFLG	011	BIT 4	FLAG 0	NEEDLBIT
6	#	NEWIFLG	122	BIT 13	FLAG 8	NEWIBIT
7	#	NJETSFLG	015	BIT 15	FLAG	NJETSBIT
8	#	NODOFLAG	044	BIT 1	FLAG 2	NODOBIT
9	#	NOLRREAD	170	BIT 10	FLAG 11	NOLRRBIT
10	#	NORMSW	110	BIT 10	FLAG 7	NORMSBIT
11	#	NORRMON	086	BIT 4	FLAG 5	NORRMBIT
12	#	NOR29FLG	049	BIT 11	FLAG 3	NR29FBIT
13	#	NOTHROTL	078	BIT 12	FLAG 5	NOTHRBIT
14	#	NOUPFLAG	024	BIT 6	FLAG 1	NOUPFBIT
15	#	NRMNVFLG	067	BIT 8	FLAG 4	NRMNVBIT
16	#	NRMIDFLG	062	BIT 13	FLAG 4	NRMIDBIT
17	#	NRUPTFLG	071	BIT 4	FLAG 4	NRUPTBIT
18	#	NTARGFLG	102	BIT 3	FLAG 6	NTARGBIT
19	#	NWAITFLG	065	BIT 10	FLAG 4	NWAITBIT
20	#	OLDESFLG	014	BIT 1	FLAG 0	OLDESBIT
21	#	OPTNSW	038	BIT 7	FLAG 2	OPTNBIT
22	#	ORBWFLAG	054	BIT 6	FLAG 3	ORBWFBIT
23	#	ORDERSW	129	BIT 6	FLAG 8	ORDERBIT
24	#	OURRCFLG	198	BIT 12	FLAG 13	OURRCBIT
25	#	PDSPFLAG	063	BIT 12	FLAG 4	PDSPFBIT
26	#	PFRATFLG	041	BIT 4	FLAG 2	PFRATBIT
27	#	PINBRFLG	069	BIT 6	FLAG 4	PINBRBIT
28	#	PRECIFLG	052	BIT 8	FLAG 3	PRECIBIT
29	#	PRIODFLG	061	BIT 14	FLAG 1	PRIODBIT
30	#	PRONVFLG	068	BIT 7	FLAG 4	PRONVBIT
31	#	PSTHIGAT	169	BIT 11	FLAG 11	PSTHIBIT
32	#	PULSEFLG	195	BIT 15	FLAG 13	PULSES
33	#	P21FLAG	004	BIT 11	FLAG 0	P21FLBIT
34	#	P25FLAG	006	BIT 9	FLAG 0	P25FLBIT
35	#	P39/79SW	126	BIT 9	FLAG 8	P39SWBIT
36	#	QUITFLAG	145	BIT 5	FLAG 9	QUITBIT
37	#	RADMODES		FLGWRD12		
38	#	RASFLAG		FLGWRD10		
39	#	RCDUFAIL	188	BIT 7	FLAG 12	RCDUFBIT
40	#	RCDUOFLG	182	BIT 13	FLAG 12	RCDUOBIT
41	#	READLR	174	BIT 6	FLAG 11	READLBIT

EQUIVALENT FLAG NAME: ETPIFLAG

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	1
78	
79	
80	

FLAGWORD_ASSIGNMENTS

# V37FLAG	114	BIT 6	FLAG 7	V37FLBIT
# V67FLAG	112	BIT 8	FLAG 7	V67FLBIT
# V82EMFLG	118	BIT 2	FLAG 7	V82EMBIT
# XDELVFLG	037	BIT 8	FLAG 2	XDELVBIT
# XDSPFLAG	074	BIT 1	FLAG 4	XDSPBIT
# XORFLG	171	BIT 9	FLAG 11	XORFLBIT
# XOVINFLG	201	BIT 9	FLAG 13	XOVINHIB
# 3AXISFLG	084	BIT 6	FLAG 5	3AXISBIT
# 360SW	134	BIT 1	FLAG 8	360SWBIT

ASSIGNMENT AND DESCRIPTION OF FLAGWORDS

FLAGWRD0 = STATE +0 # (000-014)

(SET) (RESET)

BIT 15 FLAG 0 (S)

= 000D
= BIT15

BIT 14 FLAG 0 (S)

JSWITCH	=	001D	#	INTEGRATION OF W	INTEGRATION OF STATE
JSWCHBIT	=	BIT14	#	MATRIX	VECTOR

BIT 13 FLAG 0 (S)

MIDFLAG	=	002D	#	INTEGRATION WITH	INTEGRATION WITHOUT
			#	SECONDARY BODY AND	SOLAR PERTURBATIONS
MIDFLBIT	=	BIT13	#	SOLAR PERTURBATIONS	

BIT 12 FLAG 0 (L)

MOONFLAG	=	003D	#	MOON IS SPHERE OF	EARTH IS SPHERE OF
MOONBIT	=	BIT12	#	INFLUENCE	INFLUENCE

BIT 11 FLAG 0

P21FLAG	=	004D	#	USE BASE VECTORS	1ST PASS -- CALC-
P21FLBIT	=	BIT11	#	ALREADY CALCULATED	ULATE BASE VECTORS

BIT 10 FLAG 0

FSPASFLG	=	005D	#	FIRST PASS THROUGH	NOT FIRST PASS THRU
FSPASBIT	=	BIT10	#	REPOSITION ROUTINE	REPOSITION ROUTINE

FLAGWORD_ASSIGNMENTS

1						1
2	# BIT 9 FLAG 0 (S)					2
3	P25FLAG	=	006D	#	P25 OPERATING	3
4	P25FLBIT	=	BIT9		P25 NOT OPERATING	4
5						5
6	# BIT 8 FLAG 0 (S)					6
7	IMUSE	=	007D	#	IMU IN USE	7
8	IMUSEBIT	=	BIT8		IMU NOT IN USE	8
9						9
10	# BIT 7 FLAG 0 (S)					10
11	RNDVZFLG	=	008D	#	P20 RUNNING (RADAR	11
12	RNDVZBIT	=	BIT7	#	IN USE)	12
13						13
14	# BIT 6 FLAG 0 (S)					14
15	RRNBSW	=	009D	#	RADAR TARGET IN	15
16	RRNBBIT	=	BIT6	#	NB COORDINATES	16
17						17
18	# BIT 5 FLAG 0 (S)					18
19	LOKONSW	=	010D	#	RADAR LOCK-ON	19
20	LOKONBIT	=	BIT5	#	DESIRED	20
21						21
22	# BIT 4 FLAG 0 (S)					22
23	NEEDLFLG	=	011D	#	TOTAL ATTITUDE	23
24	NEEDLBIT	=	BIT4	#	ERROR DISPLAYED	24
25						25
26	# BIT 3 FLAG 0					26
27	FREEFLAG	=	012D	#	(USED BY P51-53 TEMP IN MANY DIFFERENT	27
28						28
29	FREEFBIT	=	BIT3	#	ROUTINES & BY LUNAR + SOLAR EPHEMERIDES)	29
30						30
31	# BIT 2 FLAG 0					31
32	R10FLAG	=	013D	#	R10 OUTPUTS DATA TO	32
33	R10FLBIT	=	BIT2	#	ALTITUDE & ALTITUDE	33
34						34
35				#	RATE METERS ONLY	35
36						36
37	# BIT 1 FLAG 0 (L)					37
38	OLDESFLG	=	014D	#	R29 GYRO CMD LOOP	38
39	OLDESBIT	=	BIT1	#	REQUESTED	39
40						40
41	FLAGWRD1	=	STATE +1	#	(015-029)	41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1412THE

1

[illegible]

FLAGWORD_ASSIGNMENTS

# BIT 5 FLAG 1 (S)				
TRACKFLG	=	025D	#	TRACKING ALLOWED
TRACKBIT	=	BIT5		TRACKING NOT ALLOWED
# BIT 4 FLAG 1				
	=	026D		
	=	BIT4		
# BIT 3 FLAG 1 (S)				
SLOPESW	=	027D	#	ITERATE WITH BIAS
			#	METHOD IN ITERATOR
SLOPEBIT	=	BIT3	#	ITERATE WITH REGULAR
				FALSI METHOD IN
# BIT 2 FLAG 1 (S)				
GUESSW	=	028D	#	NO STARTING VALUE
GUESSBIT	=	BIT2	#	FOR ITERATION
				STARTING VALUE FOR
				ITERATION EXISTS
# BIT 1 FLAG 1				
	=	029D		
	=	BIT1		
			#	OH 2009-05-15 SCAN DOES NOT HAVE THIS LINE
FLAGWRD2	=	STATE +2	#	(030-044)
			#	(SET)
				(RESET)
# BIT 15 FLAG 2 (S)				
DRIFTFLG	=	030D	#	T3RUPT CALLS GYRO
DRFTBIT	=	BIT15	#	COMPENSATION
				T3RUPT DOES NO GYRO
				COMPENSATION
# BIT 14 FLAG 2 (S)				
SRCHOPTN	=	031D	#	RADAR IN AUTOMATIC
SRCHOBIT	=	BIT14	#	SEARCH OPTION (R24)
				RADAR NOT IN AUTO-
				MATIC SEARCH OPTION
# BIT 13 FLAG 2 (S)				
ACMODFLG	=	032D	#	MANUAL ACQUISITION
ACMODBIT	=	BIT13	#	BY RENDEZVOUS RADAR
				AUTO ACQUISITION
				BY RENDEZVOUS RADAR
# BIT 12 FLAG 2 (S)				
LOSCMFLG	=	033D	#	LINE OF SIGHT BEING
			#	COMPUTED (R21)
LOSCMBIT	=	BIT12		LINE OF SIGHT NOT
				BEING COMPUTED

# BIT 11 FLAG 2 (S)					
STEERSW	=	034D	#	SUFFICIENT THRUST	INSUFFICIENT THRUST
STEERBIT	=	BIT11	#	IS PRESENT	IS PRESENT
# BIT 10 FLAG 2 (S)					
	=	035D	#	# OH 2009-05-15 THESE TWO LINE DON'T APPEAR IN SCAN	
	=	BIT10			
# BIT 9 FLAG 2 (S)					
IMPULSW	=	036D	#	MINIMUM IMPULSE	STEERING BURN (NO
			#	BURN (CUTOFF TIME	CUTOFF TIME YET
IMPULBIT	=	BIT9	#	SPECIFIED)	AVAILABLE)
# BIT 8 FLAG 2 (S)					
XDELVFLG	=	037D	#	EXTERNAL DELTAV VG	LAMBERT (AIMPOINT)
XDELVBIT	=	BIT8	#	COMPUTATION	VG COMPUTATION
# BIT 7 FLAG 2 (S)					
ETPIFLAG	=	038D	#	ELEVATION ANGLE	TPI TIME SUPPLIED
			#	SUPPLIED FOR	FOR P34,74 TO COMPUTE
ETPIBIT	=	BIT7	#	P34,74	ELEVATION
# BIT 7 FLAG 2 (L)					
OPTNSW	=	ETPIFLAG	#	SOI PHASE OF P38/78	SOR PHASE OF P38/78
OPTNBIT	=	BIT7			
# BIT 6 FLAG 2 (S)					
FINALFLG	=	039D	#	LAST PASS THROUGH	INTERIM PASS THROUGH
			#	RENDEZVOUS PROGRAM	RENDEZVOUS PROGRAM
FINALBIT	=	BIT6	#	COMPUTATIONS	COMPUTATIONS
# BIT 5 FLAG 2 (S)					
AVFLAG	=	040D	#	LEM IS ACTIVE	CSM IS ACTIVE
AVFLBIT	=	BIT5	#	VEHICLE	VEHICLE
# BIT 4 FLAG 2 (S)					
PFRATFLG	=	041D	#	PREFERRED ATTITUDE	PREFERRED ATTITUDE
PFRATBIT	=	BIT4	#	COMPUTED	NOT COMPUTED
# BIT 3 FLAG 2 (S)					

# FLAGWORD ASSIGNMENTS						PAGE	10
1							1
2	CALCMAN3	=	042D	#	NO FINAL ROLL	FINAL ROLL IS	2
3	CALC3BIT	=	BIT3	#		NECESSARY	3
4							4
5	# BIT 2 FLAG 2 (S)						5
6	CALCMAN2	=	043D	#	PERFORM MANEUVER	BYPASS STARTING	6
7	CALC2BIT	=	BIT2	#	STARTING PROCEDURE	PROCEDURE	7
8							8
9	# BIT 1 FLAG 2 (S)						9
10	NODOFLAG	=	044D	#	V37 NOT PERMITTED	V37 PERMITTED	10
11	NODOBIT	=	BIT1				11
12							12
13	FLAGWRD3	=	STATE +3	#	(045-059)		13
14				#	(SET)	(RESET)	14
15							15
16							16
17	# BIT 15 FLAG 3						17
18		=	045D	#			18
19		=	BIT15	#	OH 2009-05-15 THIS LINE IS NOT IN SCANS		19
20							20
21	# BIT 14 FLAG 3 (S)						21
22	GLOKFAIL	=	046D	#	GIMBAL LOCK HAS	NOT IN GIMBAL LOCK	22
23	GLOKFBIT	=	BIT14	#	OCCURRED		23
24							24
25	# BIT 13 FLAG 3 *** PROTECTED FROM FRESH START ***						25
26	REFSMFLG	=	047D	#	REFSMMAT GOOD	REFSMMAT NO GOOD	26
27	REFSMBIT	=	BIT13				27
28							28
29	# BIT 12 FLAG 3 (S)						29
30	LUNAFLAG	=	048D	#	LUNAR LAT-LONG	EARTH LAT-LONG	30
31	LUNABIT	=	BIT12				31
32							32
33	# BIT 11 FLAG 3 (L)						33
34	NOR29FLG	=	049D	#	R29 NOT ALLOWED	R29 ALLOWED (RR DES-	34
35	NR29FBIT	=	BIT11	#		IGNATED POWERED FLT)	35
36							36
37	# BIT 10 FLAG 3 (S)						37
38	VFLAG	=	050D	#	LESS THAN TWO STARS	TWO STARS IN FIELD	38
39	VFLAGBIT	=	BIT10	#	IN FIELD OF VIEW	OF VIEW	39
40							40
41	# BIT 9 FLAG 3 (S)						41
42	R04FLAG	=	051D	#	ALARM 521	ALARM 521 ALLOWED	42
43				#	SUPPRESSED		43
44							44
45							45
46							46
47							47
48							48
49							49
50							50
51							51
52							52
53							53
54							54
55							55
56							56
57							57
58							58
59							59
60							60

1	# FLAGWORD ASSIGNMENTS					1
2	R04FLBIT	=	BIT9			2
3						3
4	# BIT 9 FLAG 3 (L)					4
5	READRFLG	=	R04FLAG	#	READING RR DATA	5
6	READRBIT	=	BIT9	#	PURSUANT TO R29	6
7						7
8	# BIT 8 FLAG 3 (S)					8
9	PRECIFLG	=	052D	#	NORMAL INTEGRATION	9
10				#	IN P00	10
11	PRECIBIT	=	BIT8	#		11
12						12
13	# BIT 7 FLAG 3 (S)					13
14	CULTFLAG	=	053D	#	STAR OCCULTED	14
15	CULTBIT	=	BIT7		STAR NOT OCCULTED	15
16						16
17	# BIT 6 FLAG 3 (S)					17
18	ORBWFLAG	=	054D	#	W MATRIX VALID FOR	18
19	ORBWBIT	=	BIT6	#	ORBITAL NAVIGATION	19
20						20
21	# BIT 5 FLAG 3 (S)					21
22	STATEFLG	=	055D	#	PERMANENT STATE	22
23	STATEBIT	=	BIT5	#	VECTOR UPDATED	23
24						24
25	# BIT 4 FLAG 3 (S)					25
26	INTYPFLG	=	056D	#	CONIC INTEGRATION	26
27	INTYPBIT	=	BIT4		ENCKE INTEGRATION	27
28						28
29	# BIT 3 FLAG 3 (S)					29
30	VINTFLAG	=	057D	#	CSM STATE VECTOR	30
31	VINTFBIT	=	BIT3	#	BEING INTEGRATED	31
32						32
33	# BIT 2 FLAG 3 (S)					33
34	D6OR9FLG	=	058D	#	DIMENSION OF W IS 9	34
35	D6OR9BIT	=	BIT2	#	FOR INTEGRATION	35
36						36
37	# BIT 1 FLAG 3 (S)					37
38	DIM0FLAG	=	059D	#	W MATRIX IS TO BE	38
39	DIM0BIT	=	BIT1	#	USED	39
40						40
41	FLAGWRD4	=	STATE +4	#	(060-074)	41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

[illegible]

1						1
2						2
3	PRONVBIT	=	BIT7	#	KEYBOARD WHEN	KEYBOARD WHEN
4				#	PRIORITY DISPLAY	PRIORITY DISPLAY
5				#	INITIATED	INITIATED
6	# BIT 6 FLAG 4 (S)					6
7	PINBRFLG	=	069D	#	ASTRONAUT HAS	ASTRONAUT HAS NOT
8				#	INTERFERED WITH	INTERFERED WITH
9	PINBRBIT	=	BIT6	#	EXISTING DISPLAY	EXISTING DISPLAY
10						10
11	# BIT 5 FLAG 4 (S)					11
12	MRUPTFLG	=	070D	#	MARK DISPLAY	MARK DISPLAY NOT
13				#	INTERRUPTED BY	INTERRUPTED BY
14	MRUPTBIT	=	BIT5	#	PRIORITY DISPLAY	PRIORITY DISPLAY
15						15
16	# BIT 4 FLAG 4 (S)					16
17	NRUPTFLG	=	071D	#	NORMAL DISPLAY	NORMAL DISPLAY NOT
18				#	INTERRUPTED BY	INTERRUPTED BY
19	NRUPTBIT	=	BIT4	#	PRIORITY OR MARK	PRIORITY OR MARK
20				#	DISPLAY	DISPLAY
21						21
22	# BIT 3 FLAG 4 (S)					22
23	MKOVFLAG	=	072D	#	MARK DISPLAY OVER	NO MARK DISPLAY OVER
24	MKOVBIT	=	BIT3	#	NORMAL	NORMAL
25						25
26	# BIT 2 FLAG 4					26
27						27
28				#	OH 2009-05-15 NOT IN SCAN.	
29						29
30						30
31	# BIT 1 FLAG 4 (S)					31
32	XDSPFLAG	=	074D	#	MARK DISPLAY NOT	NO SPECIAL MARK
33	XDSPBIT	=	BIT1	#	TO BE INTERRUPTED	INFORMATION
34						34
35	FLAGWRD5	=	STATE +5	#	(075-089)	
36				#	(SET)	(RESET)
37						37
38	# BIT 15 FLAG 5 (S)					38
39	DSKYFLAG	=	075D	#	DISPLAYS SENT TO	NO DISPLAYS TO DSKY
40	DSKYFBIT	=	BIT15	#	DSKY	
41						41
42	# BIT 14 FLAG 5					42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1						1			
2	# BIT 13 FLAG 5 (S,L)					2			
3	SNUFFER	=	077D	#	U,V JETS DISABLED	U,V JETS ENABLED	3		
4						#	DURING DPS	DURING DPS	4
5	SNUFFBIT	=	BIT13	#	BURNS (V65)	BURNS (V75)	5		
6								6	
7	# BIT 12 FLAG 5 (S)							7	
8	NOTHROTL	=	078D	#	INHIBIT FULL	PERMIT FULL THROTTLE	8		
9	NOTHRBIT	=	BIT12	#	THROTTLE		9		
10								10	
11	# BIT 11 FLAG 5 (S,L)							11	
12	R77FLAG	=	079D	#	R77 IS ON,	R77 IS NOT ON.	12		
13						#	SUPPRESS ALL RADAR		13
14						#	ALARMS AND TRACKER		14
15	R77FLBIT	=	BIT11	#	FAILS		15		
16								16	
17	# BIT 10 FLAG 5 (S)							17	
18	RNGSCFLG	=	080D	#	SCALE CHANGE HAS	NO SCALE CHANGE HAS	18		
19						#	OCCURRED DURING	OCCURRED DURING	19
20	RNGSCBIT	=	BIT10	#	RR READING	RR READING	20		
21								21	
22	# BIT 9 FLAG 5 (S)							22	
23	DMENFLG	=	081D	#	DIMENSION OF W IS 9	DIMENSION OF W IS 6	23		
24	DMENFBIT	=	BIT9	#	FOR INCORPORATION	FOR INCORPORATION	24		
25								25	
26	# BIT 8 FLAG 5 (S)							26	
27								27	
28								28	
29								29	
30	# BIT 7 FLAG 5 (S)							30	
31	ENGONFLG	=	083D	#	ENGINE TURNED ON	ENGINE TURNED OFF	31		
32	ENGONBIT	=	BIT7	#			32		
33								33	
34	# BIT 6 FLAG 5 (S)							34	
35	3AXISFLG	=	084D	#	MANEUVER SPECIFIED	MANEUVER SPECIFIED	35		
36						#	BY THREE AXES	BY ONE AXIS; R60	36
37	3AXISBIT	=	BIT6	#		CALLS VECPOINT.	37		
38								38	
39	# BIT 5 FLAG 5							39	
40								40	
41								41	
42						# OH 2009-05-15 NOT IN SCAN		42	
43	# BIT 4 FLAG 5 (S)							43	
44								44	
45								45	
46								46	
47								47	
48								48	
49								49	
50								50	
51								51	
52								52	
53								53	
54								54	
55								55	
56								56	
57								57	
58								58	
59								59	
60								60	

1412THE

FLAGWORD_ASSIGNMENTS

NORRMON	=	086D	#	BYPASS RR GIMBAL	PERFORM
NORRMBIT	=	BIT4	#	MONITOR	RR GIMBAL MONITOR
# BIT 3 FLAG 5 (S)					
SOLNSW	=	087D	#	LAMBERT DOES NOT	LAMBERT CONVERGES OR
SOLNSBIT	=	BIT3	#	CONVERGE, OR TIME-RAD	TIME-RADIUS NON-
			#	NEARLY CIRCULAR	CIRCULAR
# BIT 2 FLAG 5 (S)					
MGLVFLAG	=	088D	#	LOCAL VERTICAL	MIDDLE GIMBAL ANGLE
MGLVFBIT	=	BIT2	#	COORDINATES	COMPUTED
			#	COMPUTED	
# BIT 1 FLAG 5 (S)					
RENDWFLG	=	089D	#	W MATRIX VALID	W MATRIX INVALID
RENDWBIT	=	BIT1	#	FOR RENDEZVOUS	FOR RENDEZVOUS
			#	NAVIGATION	NAVIGATION
FLAGWRD6 = STATE +6 # (090-104)					
# (SET) (RESET)					
# BIT 15 FLAG 6 (S)					
S32.1F1	=	090D	#	DELTA V AT CSI TIME	DVT1 LESS THAN MAX
S32BIT1	=	BIT15	#	ONE EXCEEDS MAX	
# BIT 14 FLAG 6 (S)					
S32.1F2	=	091D	#	FIRST PASS OF	REITERATION OF
S32BIT2	=	BIT14	#	NEWTON ITERATION	NEWTON
# BIT 13 FLAG 6 (S)					
S32.1F3A	=	092D	#	# BIT 13 AND BIT 12 FUNCTION AS AN ORDERED	
S32BIT3A	=	BIT13	#	# PAIR (13,12) INDICATING THE POSSIBLE OC-	
			#	# CURRENCE OF 2 NEWTON ITERATIONS FOR S32.1	
			#	# IN THE PROGRAM IN THE FOLLOWING ORDER:	
# BIT 12 FLAG 6 (S)					
S32.1F3B	=	093D	#	# (0,1) (I.E. BIT 13 RESET, BIT 12 SET)	
S32BIT3B	=	BIT12	#	# = FIRST NEWTON ITERATION BEING DONE	
			#	# (0,0)= FIRST PASS OF SECOND NEWTON ITERATION	
			#	# (1,1)= 50 FT/SEC STAGE OF SECOND NEWTON ITERATION	
			#	# (1,0)= REMAINDER OF SECOND NEWTON ITERATION	
# BIT 11 FLAG 6 (S)					
	=	094D	#		
	=	BIT11	#		

FLAGWORD_ASSIGNMENTS

1						1
2	# BIT 10 FLAG 6 (S)					2
3	GMBDRVSW	=	095D	#	TRIMGIMB OVER	3
4	GMBDRBIT	=	BIT10	#	TRIMGIMB NOT OVER	4
5						5
6	# BIT 9 FLAG 6					6
7		=	096D	#		7
8		=	BIT9	#		8
9						9
10	# BIT 8 FLAG 6 (S)					10
11	MUNFLAG	=	097D	#	SERVICER CALLS	11
12	MUNFLBIT	=	BIT8	#	MUNRVG	12
13						13
14	# BIT 7 FLAG 6 (L)					14
15		=	098D	#		15
16		=	BIT7	#		16
17						17
18	# BIT 6 FLAG 6 (L)					18
19	REDFLAG	=	099D	#	LANDING SITE	19
20				#	REDESIGNATION	20
21	REDFLBIT	=	BIT6	#	PERMITTED	21
22						22
23	# BIT 5 FLAG 6					23
24		=	100D	#		24
25		=	BIT5	#	OH 2009-05-15 NOT IN SCAN	25
26						26
27	# BIT 4 FLAG 6					27
28		=	101D	#		28
29		=	BIT4	#	OH 2009-05-15 NOT IN SCAN	29
30						30
31	# BIT 3 FLAG 6 (S)					31
32	NTARGFLG	=	102D	#	ASTRONAUT DID	32
33				#	OVERWRITE DELTA	33
34	NTARGBIT	=	BIT3	#	VELOCITY AT TPI	34
35				#	OR TPM (P34,35)	35
36						36
37	# BIT 2 FLAG 6					37
38	AUXFLAG	=	103D	#	PROVIDING IDLEFLAG	38
39	AUXFLBIT	=	BIT2	#	IS NOT SET, SERV-	39
40				#	ICER WILL EXERCISE	40
41				#	DVMON ON ITS NEXT	41
42				#	PASS.	42
43				#		43
44						44
45	# BIT 1 FLAG 6 (L)					45
46	ATTFLAG	=	104D	#	LEM ATTITUDE EXISTS	46
47				#	IN MOON-FIXED	47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

14121HE

FLAGWORD_ASSIGNMENTS

# BIT 7 FLAG 7 (S)				
IDLEFLAG	=	113D	#	NO DV MONITOR
IDLEFBIT	=	BIT7	#	CONNECT DV MONITOR
# BIT 6 FLAG 7 (S)				
V37FLAG	=	114D	#	AVERAGEG (SERVICER)
V37FLBIT	=	BIT6	#	AVERAGEG (SERVICER) OFF
# BIT 5 FLAG 7 (S)				
AVEGFLAG	=	115D	#	AVERAGEG (SERVICER)
AVEGFBIT	=	BIT5	#	AVERAGEG (SERVICER) NOT DESIRED
# BIT 4 FLAG 7 (S)				
UPLOCKFL	=	116D	#	K-KBAR-K FAIL
UPLOCBIT	=	BIT4	#	NO K-KBAR-K FAIL
# BIT 3 FLAG 7 (S)				
VERIFLAG	=	117D	#	CHANGED WHEN V33E OCCURS AT END OF P27
VERIFBIT	=	BIT3	#	
# BIT 2 FLAG 7 (L,C)				
V82EMFLG	=	118D	#	MOON VICINITY
V82EMBIT	=	BIT2	#	EARTH VICINITY
# BIT 1 FLAG 7 (S)				
TFFSW	=	119D	#	CALCULATE TPERIGEE
TFFSWBIT	=	BIT1	#	CALCULATE TFF
FLAGWRD8	=	STATE +8D	#	(120-134)
			#	(SET)
				(RESET)
# BIT 15 FLAG 8 (S)				
RPQFLAG	=	120D	#	RPQ NOT COMPUTED
			#	RPQ COMPUTED
RPQFLBIT	=	BIT15	#	(RPQ = VECTOR BE- TWEEN SECONDARY BODY AND PRIMARY BODY)
# BIT 14 FLAG 8				
	=	121D	#	
	=	BIT14	#	

# BIT 13 FLAG 8 (S)					
NEWIFLG	=	122D	#	FIRST PASS THROUGH	SUCCEEDING ITERATION
NEWIBIT	=	BIT13	#	INTEGRATION	OF INTEGRATION
# BIT 12 FLAG 8 *** PROTECTED FROM FRESH START ***					
CMOONFLG	=	123D	#	PERMANENT CSM STATE	PERMANENT CSM STATE
CMOONBIT	=	BIT12	#	IN LUNAR SPHERE	IN EARTH SPHERE
# BIT 11 FLAG 8 *** PROTECTED FROM FRESH START ***					
LMOONFLG	=	124D	#	PERMANENT LM STATE	PERMANENT LM STATE
LMOONBIT	=	BIT11	#	IN LUNAR SPHERE	IN EARTH SPHERE
# BIT 10 FLAG 8 (L)					
FLUNDISP	=	125D	#	CURRENT GUIDANCE	CURRENT GUIDANCE
FLUNDBIT	=	BIT10	#	DISPLAYS INHIBITED	DISPLAYS PERMITTED
# BIT 9 FLAG 8 (L)					
P39/79SW	=	126D	#	P39/79 OPERATING	P38/78 OPERATING
P39SWBIT	=	BIT9	#		
# BIT 8 FLAG 8 *** PROTECTED FROM FRESH START ***					
SURFFLAG	=	127D	#	LM ON LUNAR SURFACE	LM NOT ON LUNAR
SURFFBIT	=	BIT8	#		SURFACE
# BIT 7 FLAG 8 (S)					
INFINFLG	=	128D	#	NO CONIC SOLUTION	CONIC SOLUTION
			#	(CLOSURE THROUGH	EXISTS
INFINBIT	=	BIT7	#	INFINITY REQUIRED)	
# BIT 6 FLAG 8 (S)					
ORDERSW	=	129D	#	ITERATOR USES 2ND	ITERATOR USES 1ST
ORDERBIT	=	BIT6	#	ORDER MINIMUM MODE	ORDER STANDARD MODE
# BIT 5 FLAG 8 (S)					
APSESW	=	130D	#	RDESIRED OUTSIDE	RDESIRED INSIDE
			#	PERICENTER-APOCENTER	PERICENTER-APOCENTER
APSESBIT	=	BIT5	#	RANGE IN TIME-RADIUS	RANGE IN TIME-RADIUS
# BIT 4 FLAG 8 (S)					
COGAFLAG	=	131D	#	NO CONIC SOLUTION --	CONIC SOLUTION
			#	TOO CLOSE TO RECTI-	EXISTS (COGA DOES NOT

FLAGWORD_ASSIGNMENTS

1	COGAFBIT	=	BIT4	#	LINEAR (COGA OVERFLWS)	OVERFLOW)
2						
3						
4	# BIT 3 FLAG 8 (S)					
5		=	132D	#		
6		=	BIT3	#	OH 2009-05-15 LINE NOT IN SCAN	
7						
8	# BIT 2 FLAG 8 (L)					
9	INITALGN	=	133D	#	INITIAL PASS THRU	SECOND PASS THRU P57
10	INITABIT	=	BIT2	#	P57	(CHECK RESET-MILLARD)
11						
12	# BIT 1 FLAG 8 (S)					
13	360SW	=	134D	#	TRANSFER ANGLE NEAR	TRANSFER ANGLE NOT
14	360SWBIT	=	BIT1	#	360 DEGREES	NEAR 360 DEGREES
15						
16						
17	FLAGWRD9	=	STATE +9D	#	(135-149)	
18						
19				#	(SET)	(RESET)
20						
21	# BIT 15 FLAG 9					
22		=	135D	#		
23		=	BIT15	#		
24						
25	# BIT 14 FLAG 9 (L)					
26	FLVR	=	136D	#	VERTICAL RISE	NON-VERTICAL RISE
27	FLVRBIT	=	BIT14	#	(ASCENT GUIDANCE)	
28						
29	# BIT 13 FLAG 9					
30		=	137D	#		
31		=	BIT13	#	OH 2009-05-15 LINE NOT IN SCAN	
32						
33	# BIT 12 FLAG 9 (L)					
34	FLPC	=	138D	#	NO POSITION CONTROL	POSITION CONTROL
35	FLPCBIT	=	BIT12	#	(ASCENT GUIDANCE)	
36						
37	# BIT 11 FLAG 9 (L)					
38	FLPI	=	139D	#	PRE-IGNITION PHASE	REGULAR GUIDANCE
39	FLPIBIT	=	BIT11	#	(ASCENT GUIDANCE)	
40						
41	# BIT 10 FLAG 9 (L)					
42	FLRCS	=	140D	#	RCS INJECTION MODE	MAIN ENGINE MODE
43	FLRCSBIT	=	BIT10	#	(ASCENT GUIDANCE)	
44						
45	# BIT 9 FLAG 9 (L)					
46						
47						
48						
49						
50						
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						

1	LETABORT	=	141D	#	ABORT PROGRAMS	ABORT PROGRAMS
2	LETABBIT	=	BIT9	#	ARE ENABLED	ARE NOT ENABLED
3						
4	# BIT 8 FLAG 9 (L)					
5	FLAP	=	142D	#	APS CONTINUED ABORT	APS ABORT IS NOT A
6				#	AFTER DPS STAGING	CONTINUATION
7	FLAPBIT	=	BIT8	#	(ASCENT GUIDANCE)	
8						
9	# BIT 7 FLAG 9 (L)					
10		=	143D			
11		=	BIT7		# OH 2009-05-15 LINE NOT IN SCAN	
12						
13						
14	# BIT 6 FLAG 9 (L)					
15						
16	ROTFLAG	=	144D	#	P70 AND P71 WILL	P70 AND P71 WILL NOT
17	ROTFLBIT	=	BIT6	#	FORCE VEHICLE	FORCE VEHICLE
18				#	ROTATION IN THE	ROTATION IN THE
19				#	PREFERRED DIRECTION	PREFERRED DIRECTION
20						
21	# BIT 5 FLAG 9 (S)					
22	QUITFLAG	=	145D	#	DISCONTINUE INTEGR.	CONTINUE INTEGRATION
23	QUITBIT	=	BIT5	#		
24						
25	# BIT 4 FLAG 9					
26		=	146D	#		
27		=	BIT4	#		
28						
29	# BIT 3 FLAG 9 (L)					
30	MID1FLAG	=	147D	#	INTEGRAT TO TDEC	INTEGRATE TO THE
31	MID1FBIT	=	BIT3	#		THEN-PRESENT TIME
32						
33	# BIT 2 FLAG 9 (L)					
34	MIDAVFLG	=	148D	#	INTEGRATION ENTERED	INTEGRATION WAS
35				#	FROM ONE OF MIDTOAV	NOT ENTERED VIA
36	MIDAVBIT	=	BIT2	#	PORTALS	MIDTOAV
37						
38	# BIT 1 FLAG 9 (S)					
39	AVEMIDSW	=	149D	#	AVETOMID CALLING	NO AVETOMID W INTEGR
40				#	FOR W.MATRIX INTEGR	ALLOW SET UP RM, VN
41	AVEMDBIT	=	BIT1	#	DON'T WRITE OVER RN,	PIPTIME
42				#	VN,PIPTIME	
43						
44	RASFLAG	EQUALS	FLGWRD10		# WAS ONLY AN INSTALL-ERASTALL FLAG	
45						
46						
47						
48						
49						
50						
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						

FLAGWORD_ASSIGNMENTS

1	FLGWRD10	=	STATE +10D	# (150-164)	
2					
3					
4				# (SET)	(RESET)
5					
6	# BIT 15 FLAG 10 (S)				
7		=	150D	#	
8		=	BIT15	#	OH 2009-05-15 LINE NOT IN SCAN
9					
10	# BIT 14 FLAG 10 (L,C)				
11	INTFLAG	=	151D	#	INTEGRATION IN
12	INTFLBIT	=	BIT14	#	INTEGRATION NOT IN
13					PROGRESS
14	# BIT 13 FLAG 10 (S,L)				
15	APSFLAG	=	152D	#	ASCENT STAGE
16	APSFLBIT	=	BIT13	#	DESCENT STAGE
17					*** PROTECTED FROM FRESH START ***
18	# BIT 12 FLAG 10				
19		=	153D	#	
20		=	BIT12	#	OH 2009-05-15 LINE NOT IN SCAN
21					
22	# BIT 11 FLAG 10				
23		=	154D	#	
24		=	BIT11	#	OH 2009-05-15 LINE NOT IN SCAN
25					
26	# BIT 10 FLAG 10				
27		=	155D	#	
28		=	BIT10	#	OH 2009-05-15 LINE NOT IN SCAN
29					
30	# BIT 9 FLAG 10				
31		=	156D	#	
32		=	BIT9	#	OH 2009-05-15 LINE NOT IN SCAN
33					
34	# BIT 8 FLAG 10				
35		=	157D	#	
36		=	BIT8	#	OH 2009-05-15 LINE NOT IN SCAN
37					
38	# BIT 7 FLAG 10 (L,C)				
39	REINTFLG	=	158D	#	INTEGRATION ROUTINE
40	REINTBIT	=	BIT7	#	INTEGRATION ROUTINE
41					TO BE RESTARTED
42	# BIT 6 FLAG 10				NOT TO BE RESTARTED
43		=	159D	#	
44		=	BIT6	#	OH 2009-05-15 LINE NOT IN SCAN
45					
46	# BIT 5 FLAG 10				
47		=	160D	#	
48		=	BIT5	#	OH 2009-05-15 LINE NOT IN SCAN
49					
50					
51					
52					
53					
54					
55					
56					
57					
58					
59					
60					

FLAGWORD_ASSIGNMENTS

1						1
2	# BIT 4 FLAG 10					2
3		=	161D	#		3
4		=	BIT4	#	OH 2009-05-15 LINE NOT IN SCAN	4
5						5
6	# BIT 3 FLAG 10					6
7		=	162D	#		7
8		=	BIT3	#	OH 2009-05-15 LINE NOT IN SCAN	8
9						9
10	# BIT 2 FLAG 10					10
11		=	163D	#		11
12		=	BIT2	#	OH 2009-05-15 LINE NOT IN SCAN	12
13						13
14	# BIT 1 FLAG 10					14
15		=	164D	#		15
16		=	BIT1	#	OH 2009-05-15 LINE NOT IN SCAN	16
17						17
18						18
19						19
20	FLGWRD11	=	STATE +11D	#	(165-179)	20
21						21
22				#	(SET) (RESET)	22
23						23
24	# BIT 15 FLAG 11 (L)(R12)					24
25	LRBYPASS	=	165D	#	BYPASS ALL LANDING DO NOT BYPASS LR	25
26	LRBYBIT	=	BIT15	#	RADAR UPDATES UPDATES	26
27						27
28	# BIT 14 FLAG 11					28
29		=	166D	#		29
30		=	BIT14	#		30
31						31
32	# BIT 13 FLAG 11					32
33		=	167D	#		33
34		=	BIT13	#		34
35						35
36	# BIT 12 FLAG 11 (L)(R12)					36
37	VXINH	=	168D	#	IF Z VELOCITY DATA UPDATE X AXIS	37
38				#	UNREASONABLE, VELOCITY	38
39	VXINHBIT	=	BIT12	#	BYPASS X VELOCITY	39
40				#	UPDATE ON NEXT PASS	40
41						41
42	# BIT 11 FLAG 11 (L)(R12)					42
43	PSTHIGAT	=	169D	#	PAST HIGATE PREHIGATE	43
44	PSTHIBIT	=	BIT11	#		44
45						45
46	# BIT 10 FLAG 11 (L)(R12)					46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

FLAGWORD_ASSIGNMENTS

NOLRREAD	=	170D	#	LANDING RADAR REPOSITIONING;	LR NOT REPOSITIONING
NOLRRBIT	=	BIT10	#	BYPASS UPDATE	
# BIT 9 FLAG 11 (L)(R12)					
XORFLG	=	171D	#	BELOW LIMIT INHIBIT X AXIS	ABOVE LIMIT DO NOT INHIBIT
XORFLBIT	=	BIT9	#	VERRIDE	
# BIT 8 FLAG 11					
LRINH	=	172D	#	LANDING RADAR UP-	LR UPDATES INHIBITED
LRINHBIT	=	BIT8	#	DATES PERMITTED BY ASTRONAUT	BY ASTRONAUT
# BIT 7 FLAG 11 (L)(R12)					
VELDATA	=	173D	#	LR VELOCITY MEASUREMENT MADE	LR VELOCITY MEASURE NOT MADE
VELDABIT	=	BIT7	#		
# BIT 6 FLAG 11 (L)(R12)					
READLR	=	174D	#	OK TO READ LR RANGE DATA	DO NOT READ LR RANGE DATA
READLBIT	=	BIT6	#		
# BIT 5 FLAG 11 (L)(R12)					
READVEL	=	175D	#	OK TO READ LR VELOCITY DATA	DO NOT READ LR VELOCITY DATA
READVBIT	=	BIT5	#		
# BIT 4 FLAG 11 (L)(R12)					
RNGEDATA	=	176D	#	LR ALTITUDE MEASUREMENT MADE	LR ALTITUDE MEASURE NOT MADE
RNGEDBIT	=	BIT4	#		
# BIT 3 FLAG 11					
SCALBAD	=	177D	#	LR LOW SCALE DISP-	LS SCALE DISCRETE
SCABBIT	=	BIT3	#	CREATE NOT PRESENT WHEN IT SHOULD	APPEARS OK
# BIT 2 FLAG 11 (L)(R12)					
VFLSHFLG	=	178D	#	LR VELOCITY FAIL LAMP SHOULD BE FLASHING	LR VEL FAIL LAMP SHOULDN'T FLASH
VFLSHBIT	=	BIT2	#		
# BIT 1 FLAG 11 (L)(R12)					

1	# FLAGWORD ASSIGNMENTS				PAGE 05		1
2	HFLSHFLG	=	179D	#	LR ALTITUDE FAIL	LR ALTITUDE FAIL	2
3	HFLSHBIT	=	BIT1	#	LAMP SHOULD BE	LAMP SHOULD NOT BE	3
4				#	FLASHING	FLASHING	4
5							5
6	RADMODES	EQUALS	FLGWRD12	#	RADAR FLAG WORD		6
7							7
8	FLGWRD12	=	STATE +12D	#	(180-194)	WAS RADMODES	8
9							9
10				#	(SET)	(RESET)	10
11							11
12	# BIT 15 FLAG 12						12
13	CDESFLAG	=	180D	#	CONTINUOUS DESIG-	LGC CHECKS FOR LOCK-	13
14	CDESBIT	=	BIT15	#	NATE, LGC COMMANDS	ON WHEN ANTENNA	14
15				#	RR REGARDLESS OF	BEING DESIGNATED	15
16				#	LOCK-ON		16
17							17
18	# BIT 14 FLAG 12						18
19	REMODFLG	=	181D	#	CHANGE IN ANTENNA	NO REMODE REQUESTED	19
20	REMODBIT	=	BIT14	#	MODE BEEN REQUESTED	OR OCCURRING	20
21				#	I.E., REMODE		21
22							22
23	# BIT 13 FLAG 12						23
24	RCDUOFLG	=	182D	#	RR CDU'S BEING	RR CDU'S NOT BEING	24
25	RCDUOBIT	=	BIT13	#	ZEROED	ZEROED	25
26							26
27	# BIT 12 FLAG 12						27
28	ANTENFLG	=	183D	#	RR ANTENNA MODE IS	RR ANTENNA IN MODE 1	28
29	ANTENBIT	=	BIT12	#	MODE 2		29
30							30
31	# BIT 11 FLAG 12						31
32	REPOSMON	=	184D	#	REPOSITION MONITOR.	NO REPOSITION TAKING	32
33	REPOSBIT	=	BIT11	#	RR REPOSITION IS	PLACE	33
34				#	TAKING PLACE		34
35							35
36	# BIT 10 FLAG 12						36
37	DESIGFLG	=	185D	#	RR DESIGNATE	RR DESIGNATE NOT	37
38	DESIGBIT	=	BIT10	#	REQUESTED OR IN	REQUESTED OR IN	38
39				#	PROGRESS	PROGRESS	39
40							40
41	# BIT 9 FLAG 12						41
42	ALTSCALE	=	186D	#	LR ALTITUDE READING	LR ALTITUDE READING	42
43	ALTSCBIT	=	BIT9	#	IS ON HIGH SCALE	IS ON LOW SCALE	43
44							44
45							45
46							46
47							47
48							48
49							49
50							50
51							51
52							52
53							53
54							54
55							55
56							56
57							57
58							58
59							59
60							60

FLAGWORD_ASSIGNMENTS

1						1
2	# BIT 8 FLAG 12					2
3	LRVELFLG	=	187D	#	LR VELOCITY DATA	NO LR VELOCITY DATA
4	LRVELBIT	=	BIT8	#	FAIL	FAIL
5						6
6	# BIT 7 FLAG 12					7
7	RCDUFAIL	=	188D	#	RR CDU FAIL HAS	RR CDU FAIL OCCURRED
8	RCDUFBIT	=	BIT7	#	NOT OCCURRED	
9						10
10	# BIT 6 FLAG 12					11
11	LRPOSFLG	=	189D	#	LANDING RADAR	LR POSITION 1
12	LRPOSBIT	=	BIT6	#	POSITION 2	
13						14
14	# BIT 5 FLAG 12					15
15	LRALTFLG	=	190D	#	LR ALTITUDE DATA	NO LR ALTITUDE DATA
16	LRALTBIT	=	BIT5	#	FAIL. COULD NOT BE	FAIL
17						18
18						19
19	# BIT 4 FLAG 12					20
20	RRDATAFL	=	191D	#	RR DATA FAIL.	NO RR DATA FAIL
21	RRDATABT	=	BIT4	#	DATA COULD NOT BE	
22						21
23						22
24	# BIT 3 FLAG 12					23
25	RRRSFLAG	=	192D	#	RR RANGE READING	RR RANGE READING ON
26	RRRSBIT	=	BIT3	#	ON THE HIGH SCALE	THE LOW SCALE
27						24
28	# BIT 2 FLAG 12					25
29	AUTOMODE	=	193D	#	RR NOT IN AUTO MODE.	RR IN AUTO MODE
30	AUTOMBIT	=	BIT2	#	AUTO MODE DISCRETE	
31						26
32						27
33	# BIT 1 FLAG 12					28
34	TURNONFL	=	194D	#	RR TURN-ON SEQUENCE	NO RR TURN-ON
35	TURNONBT	=	BIT1	#	IN PROGRESS. (ZERO	SEQUENCE IN PROGRESS
36						29
37						30
38						31
39	DAPBOOLS	EQUALS	FLGWRD13	#	DIGITAL AUTOPILOT FLAGWORD	
40						32
41						33
42						34
43						35
44						36
45						37
46						38
47						39
48						40
49						41
50						42
51						43
52						44
53						45
54						46
55						47
56						48
57						49
58						50
59						51
60						52

1412THE

1

14121HE

FLAGWORD_ASSIGNMENTS

1						1
2	# BIT 6 FLAG 13					2
3	ULLAGFLG	=	204D	#	ULLAGE REQUEST BY	NO INTERNAL ULLAGE
4	ULLAGER	=	BIT6	#	MISSION PROGRAM	REQUEST
5						5
6	# BIT 5 FLAG 13					6
7	AORBSFLG	=	205D	#	P-AXIS COUPLES 7.15	P-AXIS COUPLES 4.12
8	AORBSYST	=	BIT5	#	AND 8.16 PREFERRED	AND 3.11 PREFERRED
9						9
10	# BIT 4 FLAG 13					10
11	DBSELFLG	=	206D	#	MAX DB SELECTED	MIN DB SELECTED BY
12	DBSELECT	=	BIT4	#	BY CREW (5 DEG)	CREW (0.3 DEG)
13						13
14	# BIT 3 FLAG 13					14
15	ACCOKFLG	=	207D	#	CONTROL AUTHORITY	RESTART OR FRESH ST.
16	ACCSOKAY	=	BIT3	#	VALUES FROM 1/ACCS	SINCE LAST 1/ACCS;
17				#	USABLE	OUTPUTS SUSPECT.
18						18
19	# BIT 2 FLAG 13					19
20	AUTR2FLG	=	208D	# THESE FLAGS ARE USED TOGETHER TO INDICATE		20
21	AUTRATE2	=	BIT2	# ASTRONAUT-CHOSEN KALCMANU MANEUVER RATES		21
22				# (0,0)=(BIT2,BIT1)=	0.2 DEG/SEC	22
23	# BIT 1 FLAG 13					23
24	AUTR1FLG	=	209D	# (0,1)=	0.5 DEG/SEC	24
25	AUTRATE1	=	BIT1	# (1,0)=	2.0 DEG/SEC	25
26				# (1,1)=	10.0 DEG/SEC	26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

CONVENTIONS AND NOTATIONS UTILIZED FOR ERASABLE ASSIGNMENTS.

EQUALS IS USED IN TWO WAYS. IT IS OFTEN USED TO CHAIN A GROUP
OF ASSIGNMENTS SO THAT THE GROUP MAY BE MOVED WITH THE
CHANGING OF ONLY ONE CARD. EXAMPLE:

X EQUALS START
Y EQUALS X +SIZE.X
Z EQUALS Y +SIZE.Y

(X, Y, AND Z ARE CONSECUTIVE AND BEGIN AT START.
SIZE.X AND SIZE.Y ARE THE RESPECTIVE SIZES OF X AND Y.
USUALLY NUMERIC, IE. 1, 2, 6, 18D, ETC.)

EQUALS OFTEN IMPLIES THE SHARING OF REGISTERS (DIFFERENT NAMES
AND DIFFERENT DATA). EXAMPLE:

X EQUALS Y

= MEANS THAT MULTIPLE NAMES HAVE BEEN GIVEN TO THE SAME DATA.
(THIS IS LOGICAL EQUIVALENCE, NOT SHARING.) EXAMPLE:

X = Y

THE SIZE AND UTILIZATION OF AN ERASABLE ARE OFTEN INCLUDED IN
THE COMMENTS IN THE FOLLOWING FORM: M(SIZE)N.

M REFERS TO THE MOBILITY OF THE ASSIGNMENT.
B MEANS THAT THE SYMBOL IS REFERENCED BY BASIC
INSTRUCTIONS AND THUS IS E-BANK SENSITIVE.
I MEANS THAT THE SYMBOL IS REFERENCED ONLY BY
INTERPRETIVE INSTRUCTIONS, AND IS THUS E-BANK
INSENSITIVE AND MAY APPEAR IN ANY E-BANK.
SIZE IS THE NUMBER OF REGISTERS INCLUDED BY THE SYMBOL.
N INDICATES THE NATURE OF PERMANENCE OF THE CONTENTS.
PL MEANS THAT THE CONTENTS ARE PAD LOADED.
DSP MEANS THAT THE REGISTER IS USED FOR A DISPLAY.
PRM MEANS THAT THE REGISTER IS PERMANENT. IE., IT
IS USED DURING THE ENTIRE MISSION FOR ONE
PURPOSE AND CANNOT BE SHARED.
TMP MEANS THAT THE REGISTER IS USED TEMPORARILY OR
IS A SCRATCH REGISTER FOR THE ROUTINE TO WHICH
IT IS ASSIGNED. THAT IS, IT NEED NOT BE SET
PRIOR TO INVOCATION OF THE ROUTINE NOR DOES IT
CONTAIN USEFUL OUTPUT TO ANOTHER ROUTINE. THUS

#		IT MAY BE SHARED WITH ANY OTHER ROUTINE WHICH
#		IS NOT ACTIVE IN PARALLEL
#	IN	MEANS INPUT TO THE ROUTINE AND IT IS PROBABLY
#		TEMPORARY FOR A HIGHER-LEVEL ROUTINE/PROGRAM.
#	OUT	MEANS OUTPUT FROM THE ROUTINE, PROBABLY
#		TEMPORARY FOR A HIGHER-LEVEL ROUTINE/PROGRAM.

SPECIAL REGISTERS.

A	EQUALS	0	
L	EQUALS	1	# L AND Q ARE BOTH CHANNELS AND REGISTERS
Q	EQUALS	2	
EBANK	EQUALS	3	
FBANK	EQUALS	4	
Z	EQUALS	5	# ADJACENT TO FBANK AND BBANK FOR DXCH Z
BBANK	EQUALS	6	# (DTCB) AND DXCH FBANK (DTCF). # REGISTER 7 IS A ZERO-SOURCE, USED BY ZL.
ARUPT	EQUALS	10	# INTERRUPT STORAGE
LRUPT	EQUALS	11	
QRUPT	EQUALS	12	
SAMPTIME	EQUALS	13	# SAMPLED TIME 1 & 2.
ZRUPT	EQUALS	15	# (13 AND 14 ARE SPARES.)
BANKRUPT	EQUALS	16	# USUALLY HOLDS FBANK OR BBANK.
BRUPT	EQUALS	17	# RESUME ADDRESS AS WELL.
CYR	EQUALS	20	
SR	EQUALS	21	
CYL	EQUALS	22	
EDOP	EQUALS	23	# EDITS INTERPRETIVE OPERATION CODE PAIRS.
TIME2	EQUALS	24	
TIME1	EQUALS	25	
TIME3	EQUALS	26	
TIME4	EQUALS	27	
TIME5	EQUALS	30	
TIME6	EQUALS	31	
CDUX	EQUALS	32	
CDUY	EQUALS	33	
CDUZ	EQUALS	34	
CDUT	EQUALS	35	# RENR RADAR TRUNNION CDU
CDUS	EQUALS	36	# RENR RADAR SHAFT CDU
PIPAX	EQUALS	37	
PIPAY	EQUALS	40	
PIPAZ	EQUALS	41	
Q-RHCCTR	EQUALS	42	# RHC COUNTER REGISTERS
P-RHCCTR	EQUALS	43	
R-RHCCTR	EQUALS	44	
INLINK	EQUALS	45	
RNRAD	EQUALS	46	
GYROCMD	EQUALS	47	
CDUXCMD	EQUALS	50	
CDUYCMD	EQUALS	51	
CDUZCMD	EQUALS	52	
CDUTCMD	EQUALS	53	
CDUSCMD	EQUALS	54	

ERASABLE_ASSIGNMENTS

THRUST	EQUALS	55
LEMONM	EQUALS	56
OUTLINK	EQUALS	57
ALTM	EQUALS	60

INTERPRETIVE REGISTERS ADDRESSED RELATIVE TO VAC AREA.

LVSQUARE	EQUALS	34D	# SQUARE OF VECTOR INPUT TO ABVAL AND UNIT
LV	EQUALS	36D	# LENGTH OF VECTOR INPUT TO UNIT.
X1	EQUALS	38D	# INTERPRETIVE SPECIAL REGISTER RELATIVE
X2	EQUALS	39D	# TO THE WORK AREA.
S1	EQUALS	40D	
S2	EQUALS	41D	
QPRET	EQUALS	42D	

INPUT/OUTPUT CHANNELS

*** CHANNEL ZERO IS TO BE USED IN AN INDEXED OPERATION ONLY. ***

LCHAN	EQUALS	L	
QCHAN	EQUALS	Q	
HISCALAR	EQUALS	3	
LOSCALAR	EQUALS	4	
CHAN5	EQUALS	5	
CHAN6	EQUALS	6	
SUPERBNK	EQUALS	7	# SUPER-BANK.
OUTO	EQUALS	10	
DSALMOUT	EQUALS	11	
CHAN12	EQUALS	12	
CHAN13	EQUALS	13	
CHAN14	EQUALS	14	
MNKEYIN	EQUALS	15	
NAVKEYIN	EQUALS	16	
CHAN30	EQUALS	30	
CHAN31	EQUALS	31	
CHAN32	EQUALS	32	
CHAN33	EQUALS	33	
DNTM1	EQUALS	34	
DNTM2	EQUALS	35	

END OF CHANNEL ASSIGNMENTS



ERASABLE_ASSIGNMENTS

INTERPRETIVE SWITCH BIT ASSIGNMENTS

** FLAGWORDS AND BITS NOW ASSIGNED AND DEFINED IN THEIR OWN LOG SECTION. **



ERASABLE_ASSIGNMENTS

GENERAL ERASABLE ASSIGNMENTS

SETLOC 61
INTERRUPT TEMPORARY STORAGE POOL. (11D)

(ITEMP1 THROUGH RUPTREG4)

ANY OF THESE MAY BE USED AS TEMPORARIES DURING INTERRUPT OR WITH INTERRUPT INHIBITED. THE ITEMPS SERIES
IS USED DURING CALLS TO THE EXECUTIVE AND WAITLIST -- THE RUPTREGS ARE NOT.

ITEMP1 ERASE
WAITEXIT EQUALS ITEMP1
EXECTEM1 EQUALS ITEMP1

ITEMP2 ERASE
WAITBANK EQUALS ITEMP2
EXECTEM2 EQUALS ITEMP2

ITEMP3 ERASE
RUPTSTOR EQUALS ITEMP3
WAITADR EQUALS ITEMP3
NEWPRIO EQUALS ITEMP3

ITEMP4 ERASE
LOCCTR EQUALS ITEMP4
WAITTEMP EQUALS ITEMP4

ITEMP5 ERASE
NEWLOC EQUALS ITEMP5

ITEMP6 ERASE
NEWLOC+1 EQUALS ITEMP6 # DP ADDRESS.

SETLOC 67
NEWJOB ERASE # MUST BE AT LOC 67 DUE TO WIRING.

RUPTREG1 ERASE
RUPTREG2 ERASE
RUPTREG3 ERASE
RUPTREG4 ERASE
KEYTEMP1 EQUALS RUPTREG4
DSRUPTM EQUALS RUPTREG4

FLAGWORD RESERVATIONS. (16D)

STATE ERASE +15D # +15D FLAGWORD REGISTERS

P25 RADAR STORAGE. (MAY BE UNSHARED IN E7) (TEMP OVERLAY) (2D) OVERLAYS FLGWRD 14 & 15

```
LASTYCMD      EQUALS  STATE +14D      # B(1)PRM      THESE ARE CALLED BY T4RUPT
LASTXCMD      EQUALS  LASTYCMD +1      # B(1)PRM      THEY MUST BE CONTIGUOUS, Y FIRST
```

EXEC TEMPORARIES WHICH MAY BE USED BETWEEN CCS NEWJOBS (32D) (INTB15+ THROUGH RUPTMXTM)

INTB15+ ERASE # REFLECTS 15TH BIT OF INDEXABLE ADDRESSES

DSEXIT = INTB15+ # RETURN FOR DSPIN

EXITEM = INTB15+ # RETURN FOR SCALE FACTOR ROUTINE SELECT

BLANKRET = INTB15+ # RETURN FOR 2BLANK

INTBIT15 ERASE # SIMILAR TO ABOVE.

WRDRET = INTBIT15 # RETURN FOR 5BLANK

WDRET = INTBIT15 # RETURN FOR DSPWD

DECRET = INTBIT15 # RETURN FOR PUTCOM(DEC LOAD)

21/22REG = INTBIT15 # TEMP FOR CHARIN

THE REGISTERS BETWEEN ADDRWD AND PRIORITY MUST STAY IN THE FOLLOWING ORDER FOR INTERPRETIVE TRACE.

ADDRWD ERASE # 12 BIT INTERPRETIVE OPERAND SUB-ADDRESS.

POLISH ERASE # HOLDS CADR MADE FROM POLISH ADDRESS.

UPDATRET = POLISH # RETURN FOR UPDATNN, UPDATVB

CHAR = POLISH # TEMP FOR CHARIN

ERCNT = POLISH # COUNTER FOR ERROR LIGHT RESET

DECOUNT = POLISH # COUNTER FOR SCALING AND DISPLAY (DEC)

FIXLOC ERASE # WORK AREA ADDRESS.

OVFIND ERASE # SET NON-ZERO ON OVERFLOW.

VBUF ERASE +5 # TEMPORARY STORAGE USED FOR VECTORS.

SGNON = VBUF # TEMP FOR +,- ON

NOUNTEM = VBUF # COUNTER FOR MIXNOUN FETCH

DISTEM = VBUF # COUNTER FOR OCTAL DISPLAY VERB

DECTEM = VBUF # COUNTER FOR FETCH (DEC DISPLAY VERBS)

SGNOFF = VBUF +1 # TEMP FOR +,- ON

NVTEMP = VBUF +1 # TEMP FOR NVSUB

SFTEMP1 = VBUF +1 # STORAGE FOR SF CONST HI PART (=SFTEMP2-1)

HITEMIN = VBUF +1 # TEMP FOR LOAD OF HRS,MIN,SEC

MUST = LOTEMIN-1.

CODE = VBUF +2 # FOR DSPIN

SFTEMP2 = VBUF +2 # STORAGE FOR SF CONST LO PART (=SFTEMP1+1)

LOTEMIN = VBUF +2 # TEMP FOR LOAD OF HRS,MIN,SEC

MUST = HITEMIN+1

MIXTEMP = VBUF +3 # FOR MIXNOUN DATA

SIGNRET = VBUF +3 # RETURN FOR +,- ON

ALSO MIXTEMP+1 = VBUF+4, MIXTEMP+2 = VBUF+5

BUF ERASE +2 # TEMPORARY SCALAR STORAGE.

BUF2	ERASE	+1		
INDEXLOC	EQUALS	BUF	#	CONTAINS ADDRESS OF SPECIFIED INDEX.
SWWORD	EQUALS	BUF	#	ADDRESS OF SWITCH WORD.
SWBIT	EQUALS	BUF +1	#	SWITCH BIT WITHIN THE SWITCH WORD
MPTEMP	ERASE		#	TEMPORARY USED IN MULTIPLY AND SHIFT
DMPNTEMP	=	MPTEMP	#	DMPSUB TEMPORARY
DOTINC	ERASE		#	COMPONENT INCREMENT FOR DOT SUBROUTINE
DVSIGN	EQUALS	DOTINC	#	DETERMINES SIGN OF DDV RESULT
ESCAPE	EQUALS	DOTINC	#	USED IN ARCSIN/ARCCOS.
ENTRET	=	DOTINC	#	EXIT FROM ENTER
DOTRET	ERASE		#	RETURN FROM DOT SUBROUTINE
DVNORMCT	EQUALS	DOTRET	#	DIVIDENT NORMALIZATION COUNT IN DDV.
ESCAPE2	EQUALS	DOTRET	#	ALTERNATE ARCSIN/ARCCOS SWITCH
WDCNT	=	DOTRET	#	CHAR COUNTER FOR DSPWD
INREL	=	DOTRET	#	INPUT BUFFER SELECTION (X,Y,Z, REG)
MATINC	ERASE		#	VECTOR INCREMENT IN MXV AND VXM
MAXDVSW	EQUALS	MATINC	#	+0 IF DP QUOTIENT IS NEAR ONE -- ELSE -1.
POLYCNT	EQUALS	MATINC	#	POLYNOMIAL LOOP COUNTER
DSPMMTEM	=	MATINC	#	DSPCOUNT SAVE FOR DSPMM
MIXBR	=	MATINC	#	INDICATOR FOR MIXED OR NORMAL NOUN
TEM1	ERASE		#	EXEC TEMP
POLYRET	=	TEM1		
DSREL	=	TEM1	#	REL ADDRESS FOR DSPIN
TEM2	ERASE		#	EXEC TEMP
DSMAG	=	TEM2	#	MAGNITUDE STORE FOR DSPIN
IDADDTEM	=	TEM2	#	MIXNOUN INDIRECT ADDRESS (GARBLED)
TEM3	ERASE		#	EXEC TEMP
COUNT	=	TEM3	#	FOR DSPIN
TEM4	ERASE		#	EXEC TEMP
LSTPTR	=	TEM4	#	LIST POINTER FOR GRABUSY
RELRET	=	TEM4	#	RETURN FOR RELDSP
FREERET	=	TEM4	#	RETURN FOR FREEDSP
DSPWDRET	=	TEM4	#	RETURN FOR DSPSIGN
SEPSCRET	=	TEM4	#	RETURN FOR SEPSEC
SEPMNRET	=	TEM4	#	RETURN FOR SEPMIN
TEM5	ERASE		#	EXEC TEMP
NOUNADD	=	TEM5	#	TEMP STORAGE FOR NOUN ADDRESS
NNADTEM	ERASE		#	TEMP FOR NOUN ADDRESS TABLE ENTRY
NNTYPTM	ERASE		#	TEMP FOR NOUN TYPE TABLE ENTRY
IDAD1TEM	ERASE		#	TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)
			#	MUST = IDAD2TEM-1, = IDAD3TEM-2
IDAD2TEM	ERASE		#	TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)

```
1
2                                     # MUST = IDAD1TEM+1, = IDAD3TEM-1.
3 IDAD3TEM      ERASE                # TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)
```

```
4
5 RUTMXTEM      ERASE                # MUST = IDAD1TEM+2, = IDAD2TEM+1.
6                                     # TEMP FOR SF ROUT TABLE ENTRY (MIXNN ONLY)
```

```
7 # AX*SR*T STORAGE.                (6D)
8 DEXDEX        EQUALS    TEM2        # B(1)TMP
9 DEX1          EQUALS    TEM3        # B(1)TMP
10 DEX2          EQUALS    TEM4        # B(1)TMP
11 RTNSAVER      EQUALS    TEM5        # B(1)TMP
12 TERM1TMP      EQUALS    MPAC +3     # B(2)TMP
```

```
13
14 DEXI          =          DEX1
```

```
15
16 # THE FOLLOWING 10 REGISTERS ARE USED FOR TEMPORARY STORAGE OF THE DERIVATIVE COEFFICIENT TABLE OF
17 # SUBROUTINE ROOTPSRS. THEY MUST REMAIN WITHOUT INTERFERENCE WITH ITS SUBROUTINES WHICH ARE POWRSERS (POLY).
18 # DMPSUB, DMPNSUB, SHORTMP, DDV/BDDV, ABS, AND USPRCADR.
```

```
19
20 DERCOF-8      =          MPAC -12    # ROOTPSRS DER COF N-4 HI ORDER
21 DERCOF-7      =          MPAC -11    # ROOTPSRS DER COF N-4 LO ORDER
22 DERCOF-6      =          MPAC -10    # ROOTPSRS DER COF N-3 HI ORDER
23 DERCOF-5      =          MPAC -7     # ROOTPSRS DER COF N-3 LO ORDER
24 DERCOF-4      =          MPAC -6     # ROOTPSRS DER COF N-2 HI ORDER
25 DERCOF-3      =          MPAC -5     # ROOTPSRS DER COF N-2 LO ORDER
26 DERCOF-2      =          MPAC -4     # ROOTPSRS DER COF N-1 HI ORDER
27 DERCOF-1      =          MPAC -3     # ROOTPSRS DER COF N-1 LO ORDER
28 DERCOFN       =          MPAC -2     # ROOTPSRS DER COF N, HI ORDER
29 DERCOF+1      =          MPAC -1     # ROOTPSRS DER COF N, LO ORDER
```

```
30
31 PWRPTR        =          POLISH      # ROOTPSRS POWER TABLE POINTER
32 DXCRIT        =          VBUF +2     # ROOTPSRS CRITERION FOR ENDING ITERS HI
33 DXCRIT+1      =          VBUF +3     # ROOTPSRS CRITERION FOR ENDING ITERS LOW
34 ROOTPS        =          VBUF +4     # ROOTPSRS ROOT HI ORDER
35 ROOTPS+1      =          VBUF +5     # ROOTPSRS ROOT LO ORDER
36 RETROOT       =          BUF +2      # ROOTPSRS RETURN ADDRESS OF USER
37 PWRCNT        =          MATINC      # ROOTPSRS DER TABLE LOOP COUNTER
38 DERPTR        =          TEM1       # ROOTPSRS DER TABLE POINTER
```

DYNAMICALLY ALLOCATED CORE SETS FOR JOBS (84D)

MPAC	ERASE	+6	# MULTI-PURPOSE ACCUMULATOR.
MODE	ERASE		# +1 FOR TP, +0 FOR DP, OR -1 FOR VECTOR.
LOC	ERASE		# LOCATION ASSOCIATED WITH JOB.
BANKSET	ERASE		# USUALLY CONTAINS BBANK SETTING.
PUSHLOC	ERASE		# WORD OF PACKED INTERPRETIVE PARAMETERS.
PRIORITY	ERASE		# PRIORITY OF PRESENT JOB AND WORK AREA.

	ERASE	+83D	# EIGHT SETS OF 12 REGISTERS EACH
--	-------	------	-----------------------------------

INCORP STORAGE: R22 (N29) (SHARES WITH FOLLOWING SECTION) (4D)

R22DISP	EQUALS	TIME2SAV	# I(4) N49 DISPLAY OF DELTA R AND DELTA V
---------	--------	----------	---

STANDBY VERB ERASABLES. REDOCTR BEFORE THETADS. (14D)

TIME2SAV	ERASE	+1	
SCALSAVE	ERASE	+1	
REDOCTR	ERASE		# CONTAINS NUMBER OF RESTARTS
THETAD	ERASE	+2	
CPHI	=	THETAD	# 0 DESIRED GIMBAL ANGLES
CTHETA	=	THETAD +1	# I FOR
CPSI	=	THETAD +2	# M MANEUVER
DELV	ERASE	+5	
DELVX	=	DELV	
DELVY	=	DELV +2	
DELVZ	=	DELV +4	

DOWNLINK STORAGE. (28D)

DNLSTADR	EQUALS	DNLSTCOD	
----------	--------	----------	--

DNLSTCOD	ERASE		# B(1)PRM DOWNLINK LIST CODE
----------	-------	--	------------------------------

DUMPCNT	ERASE		# B(1)
LATALST	ERASE	+25D	# (26D)
DNTMGOTO	EQUALS	LATALST +1	# B(1)

TMINDEX	EQUALS	DNTMGOTO +1	# B(1)
DUMPLOC	EQUALS	TMINDEX	# CONTAINS ECADR OF AGC DP WORD BEING DUMPED

			# AND COUNT OF COMPLETE DUMPS ALREADY
			# SENT.
DNQ	EQUALS	TMINDEX +1	# B(1)
DNTMBUFF	EQUALS	DNQ +1	# B(22)PRM DOWNLINK SNAPSHOT BUFFER

UNSWITCHED FOR DISPLAY INTERFACE ROUTINES. (10D) FIVE MORE IN EBANK 2.

```
1 RESTREG          ERASE          # B(1)PRM FOR DISPLAY RESTARTS
2 NVWORD          ERASE
3 MARKNV          ERASE
4 NVSAVE          ERASE
5 # (RETAIN THE ORDER OF CADRFLSH TO FAILREG +2 FOR DOWNLINK PURPOSES)
6 CADRFLSH        ERASE
7 CADRMARK        ERASE
8 TEMPFLSH        ERASE
9 FAILREG          ERASE    +2      # B(3)PRM 3 ALARM CODE REGISTERS
10
11 # VAC AREAS.  -- BE CAREFUL OF PLACEMENT  --      (220D)
12
13 VAC1USE          ERASE
14 VAC1            ERASE    +42D
15 VAC2USE          ERASE
16 VAC2            ERASE    +42D
17 VAC3USE          ERASE
18 VAC3            ERASE    +42D
19 VAC4USE          ERASE
20 VAC4            ERASE    +42D
21 VAC5USE          ERASE
22 VAC5            ERASE    +42D
23
24 # WAITLIST REPEAT FLAG.      (1D)
25 RUPTAGN          ERASE
26 KEYTEMP2         =      RUPTAGN    # TEMP FOR KEYRUPT, UPRUPT
27
28 # STARALIGN ERASABLES.      (13D)
29
30 STARCODE          ERASE          # (1)
31 AOTCODE           =      STARCODE
32 STARALGN          ERASE    +11D
33 SINCDU            =      STARALGN
34 COSCDU            =      STARALGN +6
35
36 SINCDUX           =      SINCDU +4
37 SINCDUY           =      SINCDU
38 SINCDUZ           =      SINCDU +2
39 COSCDUX           =      COSCDU +4
40 COSCDUY           =      COSCDU
41 COSCDUZ           =      COSCDU +2
42
43 # PHASE TABLE AND RESTART COUNTERS      (12D)
44 -PHASE1           ERASE
```

PHASE1	ERASE		
-PHASE2	ERASE		
PHASE2	ERASE		
-PHASE3	ERASE		
PHASE3	ERASE		
-PHASE4	ERASE		
PHASE4	ERASE		
-PHASE5	ERASE		
PHASE5	ERASE		
-PHASE6	ERASE		
PHASE6	ERASE		
# A**SR*T STORAGE. (6D)			
CDUSPOT	ERASE	+5	# B(6)
CDUSPOTY	=	CDUSPOT	
CDUSPOTZ	=	CDUSPOT +2	
CDUSPOTX	=	CDUSPOT +4	
# VERB 37 STORAGE (2D)			
MINDEX	ERASE		# B(1)TMP INDEX FOR MAJOR MODE
MMNUMBER	ERASE		# B(1)TMP MAJOR MODE REQUESTED BY V37
# PINBALL INTERRUPT ACTION (1D)			
DSPCNT	ERASE		# B(1)PRM COUNTER FOR DSPOUT
# PINBALL EXECUTIVE ACTION (44D)			
DSPCOUNT	ERASE		# DISPLAY POSITION INDICATOR
DECBRNCH	ERASE		# +DEC, -DEC, OCT INDICATOR
VERBREG	ERASE		# VERB CODE
NOUNREG	ERASE		# NOUN CODE
XREG	ERASE		# R1 INPUT BUFFER
YREG	ERASE		# R2 INPUT BUFFER
ZREG	ERASE		# R3 INPUT BUFFER
XREGLP	ERASE		# LO PART OF XREG (FOR DEC CONV ONLY)
YREGLP	ERASE		# LO PART OF YREG (FOR DEC CONV ONLY)
HITEMOUT	=	YREGLP	# TEMP FOR DISPLAY OF HRS,MIN,SEC
			# MUST = LOTEMOUT-1.
ZREGLP	ERASE		# LO PART OF ZREG (FOR DEC CONV ONLY)
LOTEMOUT	=	ZREGLP	# TEMP FOR DISPLAY OF HRS,MIN,SEC
			# MUST = HITEMOUT+1
MODREG	ERASE		# MODE CODE


```
1  DSPLOCK      ERASE      # KEYBOARD/SUBROUTINE CALL INTERLOCK
2  REQRET       ERASE      # RETURN REGISTER FOR LOAD
3
4  LOADSTAT     ERASE      # STATUS INDICATOR FOR LOADTST
5  CLPASS       ERASE      # PASS INDICATOR CLEAR
6  NOUT         ERASE      # ACTIVITY COUNTER FOR DSPTAB
7
8  NOUNCADR     ERASE      # MACHINE CADR FOR NOUN
9  MONSAVE      ERASE      # N/V CODE FOR MONITOR. (= MONSAVE1-1)
10 MONSAVE1     ERASE      # NOUNCADR FOR MONITOR (MATBS1) = MONSAVE+1
11
12 MONSAVE2     ERASE      # NVMONOPT OPTIONS
13 DSPTAB        ERASE      +11D  # 0-10D, DISPLAY PANEL BUFF. 11D, C/S LTS.
14 NVQTEM       ERASE      # NVSUB STORAGE FOR CALLING ADDRESS
15
16 NVBNKTEM     ERASE      # MUST = NVBNKTEM-1.
17 # NVSUB STORAGE FOR CALLING BANK
18 # MUST = NVQTEM+1
19
20 VERBSAVE      ERASE      # NEEDED FOR RECYCLE
21 CADRSTOR     ERASE      # ENDIDLE STORAGE
22 DSPLIST       ERASE      # WAITING REG FOR DSP SYST INTERNAL USE
23
24 EXTVBACT     ERASE      # EXTENDED VERB ACTIVITY INTERLOCK
25 DSPTM1        ERASE      +2    # BUFFER STORAGE AREA 1 (MOSTLY FOR TIME)
26 DSPTM2        ERASE      +2    # BUFFER STORAGE AREA 2 (MOSTLY FOR DEG)
27
28
29 DSPTMX        EQUALS    DSPTM2 +1  # B(2) S-S DISPLAY BUFFER FOR EXT. VERBS
30 NORMTEM1     EQUALS    DSPTM1    # B(3)DSP NORMAL DISPLAY REGISTERS.
31
32 # DISPLAY FOR EXTENDED VERBS (V82, R04(V62), V41(N72) ) (2D)
33
34 OPTIONX      EQUALS    DSPTMX     # (2) EXTENDED VERB OPTION CODE
35
36 # TBASES AND PHSPRDT S.          (12D)
37
38 TBASE1       ERASE
39 PHSPRDT1     ERASE
40
41 TBASE2       ERASE
42 PHSPRDT2     ERASE
43
44 TBASE3       ERASE
45 PHSPRDT3     ERASE
46
47 TBASE4       ERASE
48 PHSPRDT4     ERASE
49
50 TBASE5       ERASE
51 PHSPRDT5     ERASE
52
53 TBASE6       ERASE
54 PHSPRDT6     ERASE
55
56 # UNSWITCHED FOR DISPLAY INTERFACE ROUTINES.    (6D)
```


ERASABLE_ASSIGNMENTS

1					1
2	NVWORD1	ERASE		# B(1) PROBABLY FOR DISPLAY DURING SERVICER	2
3	EBANKSAV	ERASE			3
4	MARKEBAN	ERASE			4
5	EBANKTEM	ERASE			5
6	MARK2PAC	ERASE			6
7	R1SAVE	ERASE			7
8					8
9	# IMU COMPENSATION UNSWITCHED ERASABLE. (1D)				9
10					10
11	1/PIPADT	ERASE			11
12					12
13	# SINGLE PRECISION SUBROUTINE TEMPORARIES (2D)				13
14					14
15	TEMK	ERASE	# (1)		15
16	SQ	ERASE	# (1)		16
17					17
18	# UNSWITCHED RADAR ERASABLE				18
19					19
20	SAMPLIM	ERASE			20
21	SAMPLSUM	ERASE	+3		21
22	TIMEHOLD	ERASE	+1		22
23	RRTARGET	EQUALS	SAMPLSUM	# HALF U IT VECTOR IN SM OR NB AXES.	23
24	TANG	ERASE	+1	# DESIRE TRUNNION AND SHAFT ANGLES.	24
25	MODEA	EQUALS	TANG		25
26	MODEB	ERASE	+1	# DODES LOBBERS TANG +2.	26
27	NSAMP	EQUALS	MODEB		27
28	DESRET	ERASE			28
29	OLDATAGD	EQUALS	DESRET	# USED IN DATA READING ROUTINES.	29
30	DESCOUNT	ERASE			30
31					31
32	# *****	P22	*****	(6D)	32
33					33
34	RSUBC	EQUALS	RRTARGET	# I(6) S-S CSM POSITION VECTOR.	34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

UNSWITCHED FOR ORBIT INTEGRATION (21D)

TDEC	ERASE	+20D	# I(2)
COLREG	EQUALS	TDEC +2	# I(1)
LAT	EQUALS	COLREG +1	# I(2)
LONG	EQUALS	LAT +2	# I(2)
ALT	EQUALS	LONG +2	# I(2)
YV	EQUALS	ALT +2	# I(6)
ZV	EQUALS	YV +6	# I(6)

MISCELLANEOUS UNSWITCHED. (20D)

P40/RET	ERASE	# (WILL BE PUT IN E6 WHEN THERE IS ROOM)	
GENRET	ERASE	# B(1) R61 RETURN CADR.	
OPTION1	ERASE	# B(1) NOUN 06 USES THIS	
OPTION2	ERASE	# B(1) NOUN 06 USES THIS	
OPTION3	ERASE	# B(1) NOUN 06 USES THIS	
LONGCADR	ERASE +1	# B(2) LONGCALL REGISTER	
LONGBASE	ERASE +1		
LONGTIME	ERASE +1	# B(2) LONGCALL REGISTER	
CDUTEMPX	ERASE	# B(1)TMP	
CDUTEMPY	ERASE	# B(1)TMP	
CDUTEMPZ	ERASE	# B(1)TMP	
PIPATMPX	ERASE	# B(1)TMP	
PIPATMPY	ERASE	# B(1)TMP	
PIPATMPZ	ERASE	# B(1)TMP	
DISPDEX	ERASE	# B(1)	
TEMPR60	ERASE	# B(1)	
PRIOTIME	ERASE	# B(1)	

P27 (UPDATE PROGRAM) STORAGE (26D)

UPVERBSV	ERASE	# B(1) UPDATE VERB ATTEMPTED.	
UPTMP	ERASE +24D	# B(1)TMP SCRATCH	
INTWAK1Q	EQUALS UPTMP	# (BORROWS UPTMP REGISTERS)	
# RETAIN THE ORDER OF COMPNUMB THRU UPBUFF +19D FOR DOWNLINK PURPOSES.			
COMPNUMB	EQUALS UPTMP +1	# B(1)TMP NUMBER OF ITEMS TO BE UPLINKED	
UPOLDMOD	EQUALS COMPNUMB +1	# B(1)TMP INTERRUPTD PROGRAM MM	
UPVERB	EQUALS UPOLDMOD +1	# B(1)TMP VERB NUMBER	
UPCOUNT	EQUALS UPVERB +1	# B(1)TMP UPBUFF INDEX	
UPBUFF	EQUALS UPCOUNT +1	# B(20D)	

SPECIAL DEFINITION FOR SYSTEM TEST ERASABLE PGMS. (2D)

EBUF2	EQUALS UPTMP	# B(2) FOR EXCLUSIVE USE OF SYSTEM TEST.	
-------	--------------	--	--

PERM STATE VECTORS FOR BOOST AND DOWNLINK -- WHOLE MISSION -- (14D)

RN	ERASE	+5	# B(6)PRM
VN	ERASE	+5	# B(6)PRM
PIPTIME	ERASE	+1	# B(2)PRM (MUST BE FOLLOWED BY GDT/2)

SERVICER -- MUST FOLLOW PIPTIME -- (19D)

GDT/2	ERASE	+19D	# B(6)TMP	** MUST FOLLOW PIPTIME **
MASS	EQUALS	GDT/2 +6	# B(2)	
WEIGHT/G	=	MASS		
ABDELV	EQUALS	MASS +2	# (KALCMANU STORAGE)	
PGUIDE	EQUALS	ABDELV +1	# (2)	
DVTHRUSH	EQUALS	PGUIDE +2	# (1)	
AVEGEXIT	EQUALS	DVTHRUSH +1	# (2)	
AVGEXIT	=	AVEGEXIT		
TEMX	EQUALS	AVEGEXIT +2	# (1)	
TEMY	EQUALS	TEMX +1	# (1)	
TEMZ	EQUALS	TEMY +1	# (1)	
PIPAGE	EQUALS	TEMZ +1	# B(1)	
OUTROUTE	EQUALS	PIPAGE +1	# B(1)	

PERMANENT LEM DAP STORAGE (12D)

CH5MASK	ERASE	# B(1)PRM
CH6MASK	ERASE	# B(1)PRM JET FAILURE MASK.
DTHETASM	ERASE +5	# (6)
SPNDX	ERASE	# B(1)
RCSFLAGS	ERASE	# AUTOPILOT FLAG WORD

BIT ASSIGNMENTS:
1) ALTERYZ SWITCH (ZEROOR1)
2) NEEDLER SWITCH
3) NEEDLER SWITCH
4) NEEDLER SWITCH
5) NEEDLER SWITCH
9) JUST-IN-DETENT SWITCH
10) PBIT -- MANUAL CONTROL SWITCH
11) QRBIT -- MANUAL CONTROL SWITCH
12) PSKIP CONTROL (PJUMPADR)
13) 1/ACCJOB CONTROL (ACCSET)
GENADR OF NEXT LM DAP T5RUPT. * 2CADR *
BBCON OF NEXT LM DAP T5RUPT. 2CADR

T5ADR ERASE +1

ERASABLES FOR P64: OVERLAY OF DTHETASM, WHICH IS UNUSED (4D)

ZERLINA EQUALS DTHETASM # B(1) P64

```
ELVIRA      EQUALS  ZERLINA +1      # B(1)  P64
AZINCR1     EQUALS  ELVIRA  +1      # B(1)  P64
ELINCR1     EQUALS  AZINCR1 +1      # B(1)  P64
```

RCS FAILURE MONITOR STORAGE (1)

PVALVEST ERASE # B(1)PRM

KALCMANU/DAP INTERFACE (3D)

DELPEROR ERASE # B(1)PRM COMMAND LAGS.

DELQEROR ERASE # B(1)PRM

DELREROR ERASE # B(1)PRM

MODE SWITCHING ERASABLE. (9D)

RETAIN THE ORDER OF IMODES30 AND IMODES33 FOR DOWNLINK PURPOSES

IMODES30 ERASE # B(1)

IMODES33 ERASE

MODECADR ERASE +2 # B(3)PRM

IMUCADR EQUALS MODECADR

OPTCADR EQUALS MODECADR +1

RADCADR EQUALS MODECADR +2

ATTCADR ERASE +2 # B(3)PRM

ATTPRIO = ATTCADR +2

MARKSTAT ERASE

T4RUPT ERASABLE (2D)

DSRUPTSW ERASE

LGYRO ERASE # (1)

RENDEZVOUS RADAR TASK STORAGE (3D)

RRRET ERASE +2D # B(1)TMP P20'S, PERHAPS R29 & R12

RDES EQUALS RRRET +1 # B(1)TMP

RRINDEX EQUALS RDES +1 # B(1)TMP

MEASINC (4D)

WIXA ERASE # B(1)

WIXB ERASE # B(1)

ZIXA ERASE # B(1)

ZIXB ERASE # B(1)

AGS DUMMY ID WORD. (1D)

AGSWORD ERASE

SOME MISCELLANEOUS UNSWITCHED. (6D)

RATEINDX ERASE # (1) USED BY KALCMANU

DELAYLOC ERASE +2

LEMMASS ERASE # KEEP CONTIGUOUS W. CSMMASS. (1) EACH

CSMMASS ERASE

LESS IS MORE.

RENDEZVOUS AND LANDING RADAR DOWNLINK STORAGE. (7D)

(NORMALLY USED DURING P20, BUT MAY ALSO)

(BE REQUIRED FOR THE V62 SPURIOUS TEST.)

(PLEASE KEEP IN THIS ORDER)

DNRRANGE ERASE +6 # B(1)TMP

DNRRDOT EQUALS DNRRANGE +1 # B(1)TMP

DNINDEX EQUALS DNRRDOT +1 # B(1)TMP

DNLRVELX EQUALS DNINDEX +1 # B(1)TMP

DNLRVELY EQUALS DNLRVELX +1 # B(1)TMP

DNLRVELZ EQUALS DNLRVELY +1 # B(1)TMP

DNLRALT EQUALS DNLRVELZ +1 # B(1)TMP

INCORPORATION UNSWITCHED (2D)

W.IND EQUALS PIPAGE # B(1)

W.INDI EQUALS W.IND +1 # I(1)

SUBROUTINE BALLANGS OF R60.

BALLEXIT ERASE # B(1) SAVE LOCATION FOR BALLINGS SUBR EXIT

SOME LEM DAP STORAGE. (4D)

DAPDATR1 ERASE # B(1)DSP DAP CONFIG.

TEVENT ERASE +1 # B(2)DSP

DB ERASE # B(1)TMP DEAD BAND.

NOUN 87 (2D)

AZ ERASE +1D # B(1) AZ AND EL MUST BE CONTIGUOUS

ERASABLE_ASSIGNMENTS

EL	EQUALS	AZ +1D	# B(1)
# P63, P64, P65, P66, AND P67. (1D)			
WCHPHASE	ERASE		# B(1)
# ERASABLES FOR THE R2 LUNAR POTENTIAL MODEL (2D)			
E3J22R2M	ERASE		# I(1)
E32C31RM	ERASE		# I(1)
RADSKAL	ERASE	+1	# LR ALT DOPPLER BIAS: 2T/LAMBDA SCALED # AT 1/(2(7) M/CS)
SKALSKAL	ERASE		# LR ALT SCALE FACTOR RATIO: .2 NOM
END-UE	EQUALS		# NEXT UNUSED UE ADDRESS
# SELF-CHECK ASSIGNMENTS (17D)			
SELFERAS	ERASE	1357 - 1377	# *** MUST NOT BE MOVED ***
SFAIL	EQUALS	SELFERAS	# B(1)
ERESTORE	EQUALS	SFAIL +1	# B(1)
SELFRET	EQUALS	ERESTORE +1	# B(1) RETURN
SMODE	EQUALS	SELFRET +1	# B(1)
ALMCADR	EQUALS	SMODE +1	# B(2) ALARM-ABORT USER'S 2CADR
ERCOUNT	EQUALS	ALMCADR +2	# B(1)
SCOUNT	EQUALS	ERCOUNT +1	# B(3)
SKEEP1	EQUALS	SCOUNT +3	# B(1)
SKEEP2	EQUALS	SKEEP1 +1	# B(1)
SKEEP3	EQUALS	SKEEP2 +1	# B(1)
SKEEP4	EQUALS	SKEEP3 +1	# B(1)
SKEEP5	EQUALS	SKEEP4 +1	# B(1)
SKEEP6	EQUALS	SKEEP5 +1	# B(1)
SKEEP7	EQUALS	SKEEP6 +1	# B(1)

ERASABLE_ASSIGNMENTS

EBANK-3 ASSIGNMENTS

SETLOC 1400

WAITLIST TASK LISTS. (26D)

LST1	ERASE	+7	# B(8D)PRM DELTA T'S.
LST2	ERASE	+17D	# B(18D)PRM TASK 2CADR ADDRESSES.

RESTART STORAGE. (2D)

RSBBQ ERASE +1 # B(2)PRM SAVE BB AND Q FOR RESTARTS

MORE LONGCALL STORAGE. (MUST BE IN LST1'S BANK. (2D)

LONGEXIT ERASE +1 # B(2)TMP MAY BE SELDOM OVERLAYED.

PHASE-CHANGE LISTS PART II. (12D)

PHSNAME1	ERASE	# B(1)PRM
PHSBB1	ERASE	# B(1)PRM
PHSNAME2	ERASE	# B(1)PRM
PHSBB2	ERASE	# B(1)PRM
PHSNAME3	ERASE	# B(1)PRM
PHSBB3	ERASE	# B(1)PRM
PHSNAME4	ERASE	# B(1)PRM
PHSBB4	ERASE	# B(1)PRM
PHSNAME5	ERASE	# B(1)PRM
PHSBB5	ERASE	# B(1)PRM
PHSNAME6	ERASE	# B(1)PRM
PHSBB6	ERASE	# B(1)PRM

IMU COMPENSATION PARAMETERS (22D)

PBIASX	ERASE	# B(1) PIPA BIAS, PIPA SCALE FACTOR TERMS
PIPABIAS	=	PBIASX # INTERMIXED.
PIPASCFX	ERASE	

PIPASCF	=	PIPASCFX
---------	---	----------

PBIASY	ERASE
PIPASCFY	ERASE

PBIASZ	ERASE
PIPASCFZ	ERASE

NBDX	ERASE	# GYRO BIAS DRIFT
NBDY	ERASE	
NBDZ	ERASE	

```
ADIAX      ERASE      # ACCELERATION SENSITIVE DRIFT ALONG THE
ADIAZ      ERASE      # INPUT AXIS
```

```
ADSRAX     ERASE      # ACCELERATION SENSITIVE DRIFT ALONG THE
ADSRAY     ERASE      # SPIN REFERENCE AXIS
ADSRAX     ERASE
```

```
GCOMP      ERASE      +5      # CONTAINS COMPENSATING TORQUES
```

```
COMMAND    EQUALS    GCOMP
CDUIND      EQUALS    GCOMP      +3
```

```
GCOMP SW    ERASE
```

```
# STATE VECTORS FOR ORBIT INTEGRATION.      (44D)
```

```
# (DIFEQCNT THUR XKEP MUST BE IN THE SAME
# EBANK AS RRECTCSM, RRECTLEM ETC
# BECAUSE THE COPY-CYCLES (ATOPCSM,
# PTOACSM ETC) ARE EXECUTED IN BASIC.
# ALL OTHER REFERENCES TO THIS GROUP
# ARE BY INTERPRETIVE INSTRUCTIONS.)
```

```
DIFEQCNT    ERASE      +43D      # B(1)
# (UPSVFLAG...XKEP MUST BE KEPT IN ORDER)
```

```
UPSVFLAG    EQUALS    DIFEQCNT +1      # B(1)
```

```
RRECT        EQUALS    UPSVFLAG +1      # B(6)
```

```
VRECT        EQUALS    RRECT      +6      # B(6)
```

```
TET          EQUALS    VRECT      +6      # B(2)
```

```
TDELTAV      EQUALS    TET        +2      # B(6)
```

```
TNUV         EQUALS    TDELTAV    +6      # B(6)
```

```
RCV          EQUALS    TNUV       +6      # B(6)
```

```
VCV          EQUALS    RCV        +6      # B(6)
```

```
TC           EQUALS    VCV        +6      # B(2)
```

```
XKEP         EQUALS    TC         +2      # B(2)
```

```
# PERMANENT STATE VECTORS AND TIMES.
```

```
# (DO NOT OVERLAY WITH ANYTHING AFTER BOOST)
```

```
# (RRECTCSM...XKEPCSM MUST BE KEPT IN THIS ORDER)
```

```
RRECTCSM     ERASE      +5      # B(6)PRM CSM VARIABLES.
```

```
RRECTOTH      =      RRECTCSM
```

```
VRECTCSM     ERASE      +5      # B(6)PRM
```



```
1  TETCSM          ERASE  +1          # B(2)PRM
2  TETOTHER        =      TETCSM
3
4  DELTACSM        ERASE  +5          # B(6)PRM
5  NUVCSM          ERASE  +5          # B(6)PRM
6  RCVCSM          ERASE  +5          # B(6)PRM
7  VCVCSM          ERASE  +5          # B(6)PRM
8  TCCSM           ERASE  +1          # B(2)PRM
9  XKEPCSM         ERASE  +1          # B(2)PRM
10
11 # (RRECTLEM...XKEPLEM MUST BE KEPT IN THIS ORDER)
12
13 RRECTLEM         ERASE  +5          # B(6)PRM LEM VARIABLES
14 RRECTHIS        =      RRECTLEM
15 VRECTLEM         ERASE  +5          # B(6)PRM
16 TETLEM           ERASE  +1          # B(2)PRM
17 TETTHIS         =      TETLEM
18 DELTALEM         ERASE  +5          # B(6)PRM
19 NUVLEM           ERASE  +5          # B(6)PRM
20 RCVLEM           ERASE  +5          # B(6)PRM
21 VCVLEM           ERASE  +5          # B(6)PRM
22 TCLEM            ERASE  +1          # B(2)PRM
23 XKEPLEM          ERASE  +1          # B(2)PRM
24
25 X789             ERASE  +5
26 TEPHEM           ERASE  +2
27 AZO              ERASE  +1
28 -AYO             ERASE  +1
29 AXO              ERASE  +1
30
31 # STATE VECTORS FOR DOWNLINK          (12D)
32
33 R-OTHER          ERASE  +5          # B(6)PRM POS VECT (OTHER VECH) FOR DNLINK
34 V-OTHER          ERASE  +5          # B(6)PRM VEL VECT (OTHER VECH) FOR DNLINK
35
36 T-OTHER          =      TETCSM      #          TIME (OTHER VECH) FOR DNLINK
37
38 # REFSMMAT.          (18D)
39
40 REFSMMAT         ERASE  +17D        # I(18D)PRM
41
42 # ACTIVE VEHICLE CENTANG. MUST BE DISPLAYED ANYTIME (ALMOST.) (2D)
43
44 ACTCENT          ERASE  +1          # I(2) S-S CENTRAL ANGLE BETWEEN ACTIVE
45                                     # VEHICLE AT TPI TIG AND TARGET VECTOR.
46
47 # ***** USED IN CONICSEX (PLAN INERT ORIENT) *****
```

ERASABLE_ASSIGNMENTS

1	# ERASABLE_ASSIGNMENTS				1
2	TIMSUBO	EQUALS	TEPHEM	# CSEC B-42 (TRIPLE PRECISION)	2
3					3
4	# LPS20.1 STORAGE	--	ALL ARE PRM --	(9D)	4
5					5
6	LS21X	ERASE		# I(1)	6
7	LOSVEL	ERASE	+5	# I(6)	7
8	MLOSV	ERASE	+1	# I(2) MAGNITUDE OF LOS. METERS B-29	8
9					9
10	# ***** P22 ***** (OVERLAYS LPS 20.1 STORAGE)			(6D)	10
11	VSUBC	EQUALS	LOSVEL	# I(6) S-S CSM VELOCITY VECTOR	11
12					12
13	# PADLOADED ERASABLES FOR P20/P22			(6D)	13
14					14
15	RANGEVAR	ERASE	+1	# I(2) RR RANGE ERROR VARIANCE	15
16	RATEVAR	ERASE	+1	# I(2) RR RANGE RATE ERROR VARIANCE	16
17	RVARMIN	ERASE		# I(1) MINIMUM RANGE ERROR VARIANCE	17
18	VVARMIN	ERASE		# I(1) MINIMUM RANGE-RATE ERROR VARIANCE	18
19					19
20	# P32-P33 STORAGE			(2D)	20
21					21
22	TCDH	ERASE	+1	# I(2) T2 CDH TIME IN CS. (ALSO DOWNLINKED)	22
23					23
24	END-E3	EQUALS	1777	# ** LAST LOCATION USED IN E3 **	24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

ERASABLE_ASSIGNMENTS

EBANK-4 ASSIGNMENTS

SETLOC 2000

E4 IS, FOR THE MOST PART RESERVED FOR PAD LOADED AND UNSHARABLE ERASE.

AMEMORY EQUALS

P20 STORAGE. -- PAD LOADED -- (6D)

WRENDPOS	ERASE	# B(1)PL	KM*2(-7)
WRENDVEL	ERASE	# B(1)PL	KM(-1/2)*2(11)
WSHAFT	ERASE	# B(1)PL	KM*2(-7)
WTRUN	ERASE	# B(1)PL	KM*2(-7)
RMAX	ERASE	# B(1)PL	METERS*2(-19)
VMAX	ERASE	# B(1)PL	M/CSEC*2(-7)

LUNAR SURFACE NAVIGATION (2D)

WSURFPOS	ERASE	# B(1)PL
WSURFVEL	ERASE	# B(1)PL

P22 STORAGE. -- PAD LOADED -- (2D)

SHAFTVAR	ERASE	# B(1)PL	RAD SQ*2(12)
TRUNVAR	ERASE	# B(1)PL	RAD SQ*2(10)

CONISEX STORAGE. -- PAD LOADED --

504LM ERASE +5 # I(6) MOON LIBRATION VECTOR

V47 (R47) AGS INITIALIZATION STORAGE. -- PAD LOADED -- (2D)

AGSK ERASE +1

LUNAR LANDING STORAGE. -- PAD LOADED -- (6D)

RLS ERASE +5 # I(6) LANDING SITE VECTOR -- MOON REF

INTEGRATION STORAGE. (102D)

PBODY ERASE +101D # I(1)

```
1  ALPHAV      EQUALS  PBODY +1      # I(6)
2  BETAV      EQUALS  ALPHAV +6      # I(6)
3
4  PHIV       EQUALS  BETAV +6      # I(6)
5  PSIV       EQUALS  PHIV +6       # I(6)
6  FV         EQUALS  PSIV +6       # I(6)  PERTURBING ACCELERATIONS
7
8  ALPHAM      EQUALS  FV +6        # I(2)
9  BETAM      EQUALS  ALPHAM +2     # I(2)
10 TAU.        EQUALS  BETAM +2     # I(2)
11
12 DT/2        EQUALS  TAU. +2      # I(2)
13 H           EQUALS  DT/2 +2      # I(2)
14 GMODE       EQUALS  H +2         # I(1)
15
16 IRETURN     EQUALS  GMODE +1     # I(1)
17 NORMGAM     EQUALS  IRETURN +1   # I(1)
18 RPQV        EQUALS  NORMGAM +1
19
20 ORIGEX      EQUALS  RPQV +6      # I(1)
21 KEPRTN      EQUALS  ORIGEX       # I(1)
22 RQVV        EQUALS  ORIGEX +1    # I(6)
23
24 RPSV        EQUALS  RQVV +6      # I(6)
25 XKEPNEW     EQUALS  RPSV +6      # I(2)
26 VECTAB      EQUALS  XKEPNEW +2   # I(36D)
27 VECTABND    EQUALS  VECTAB +35D  # END MARK
```

```
28
29 # THESE PROBABLY CAN SHARE MID-COURSE VARIABLES.      (6D)
```

```
30
31 VACX        EQUALS  VECTAB +6     # I(2)
32 VACY        EQUALS  VACX +2       # I(2)
33 VACZ        EQUALS  VACY +2       # I(2)
```

```
34
35 # SERVICER STORAGE (USED BY ALL POWERED FLIGHT PROGS.) (18D)
```

```
36
37 XNBPIP      EQUALS  VECTAB +12D   # I(6)
38 YNBPIP      EQUALS  XNBPIP +6    # I(6)
39 ZNBPIP      EQUALS  YNBPIP +6    # I(6)
```

```
40
41 # SOME VERB 82 STORAGE      (4D)
```

```
42
43 HAPOX       EQUALS  RQVV +4       # I(2)
44 HPERX       EQUALS  HAPOX +2      # I(2)
```

```
45
46 # V82 STORAGE      (6D)
```

```
47
48 VONE'       EQUALS  VECTAB +30D   # I(T)TMP NORMAL VELOCITY VONE / SQRT. MU
```

```
49
50 # R32(V83) STORAGE. -- SHARES WITH INTEGRATION STORAGE --      (28D)
```

1					1
2	BASETHV	EQUALS	RPQV	# I(6) BASE VEL VECTOR THIS VEH	2
3					3
4	BASETIME	EQUALS	RQVV	# I(2) TIME ASSOC WITH BASE VECS	4
5	ORIG	EQUALS	RQVV +2	# I(1) =0 FOR EARTH =2 FOR MOON	5
6	STATEXIT	EQUALS	RQVV +3	# I(1) STQ ADDRESS FOR STATEXTP	6
7	BASEOTV	EQUALS	RQVV +4	# I(6) BASE VEL VECTOR OTHER VEH	7
8					8
9	BASEOTP	EQUALS	VECTAB +6	# I(6) BASE POS VECTOR OTHER VEH	9
10					10
11	BASETHP	EQUALS	VECTAB +30D	# I(6) BASE POS VECTOR THIS VEH	11
12					12
13	# KEPLER STORAGE. (KEPLER IS CALLED BY PRECISION INTEGRATION AND (2D)				13
14	# CONICS)				14
15					15
16	EPSILONT	ERASE	+1	# I(2)	16
17					17
18	# VERB 83 STORAGE (18D)				18
19					19
20	RANGE	ERASE	+17D	# I(2)DSP NOUN 54 DISTANCE TO OPTICAL SUBJ	20
21	RRATE	EQUALS	RANGE +2	# I(2)DSP NOUN 54 RATE OF APPROACH	21
22	RTHETA	EQUALS	RRATE +2	# I(2)DSP NOUN 54.	22
23	RONE	EQUALS	RTHETA +2	# I(6)TMP VECTOR STORAGE. (SCRATCH)	23
24	VONE	EQUALS	RONE +6	# I(6)TMP VECTOR STORAGE. (SCRATCH)	24
25					25
26	# VERB 67 STORAGE				26
27					27
28	WWPOS	=	RANGE	# NOUN 99 (V67)	28
29	WWVEL	=	RRATE	# NOUN 99 (V67)	29
30	WWBIAS	=	RTHETA	# NOUN 99 (V67)	30
31					31
32	# V82 STORAGE. (CANNOT OVERLAY RONE OR VONE) (11D) TWO SEPARAT LOCATIONS				32
33					33
34	V82FLAGS	EQUALS	VECTAB +6	# (1) FOR V82 BITS.	34
35	TFF	EQUALS	V82FLAGS +1	# I(2)	35
36	-TPER	EQUALS	TFF +2	# I(2)	36
37					37
38	HPERMIN	EQUALS	RANGE	# I(2) SET TO 300KFT FOR SR30.1	38
39	RPADTEM	EQUALS	HPERMIN +2	# I(2) PAD OR LANDING RADIUS FOR SR30.1	39
40	TSTART82	EQUALS	RPADTEM +2	# I(2) TEMP TIME STORAGE VOR V82.	40
41					41
42	# VARIOUS DISPLAY REGISTERS (6D) NOUN 84; P76				42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

```
1 DELVOV          ERASE   +5D          # (6)
2
3
4 # ALIGNMENT PLANETARY -- INERTIAL TRANSFORMATION STORAGE.      (18D)
5
6 #
7     UNSHARED WHILE LM ON LUNAR SURFACE.
8
9 GSAV            ERASE   +17D          # I(6)
10 YNBSAV          EQUALS  GSAV +6       # I(6)
11 ZNBSAV          EQUALS  YNBSAV +6     # I(6)
12
13 # KALCMANU STORAGE, CAN OVERLAY GSAV.      (18D)
14
15 MFS            EQUALS  GSAV          # I(18)
16 MFI            EQUALS  MFS           # I
17
18 KEL            EQUALS  MFS           # I(18)
19 E01            EQUALS  MFS           # I(6)
20 E02            EQUALS  E01 +6        # I(6)
21
22 # LR VEL BEAM VECTORS.      (26D)
23
24 # CAN OVERLAY GSAV WITH CARE, USED DURING POWERED DESCENT ONLY.
25
26 VZBEAMNB       EQUALS  GSAV          # I(6) LR VELOCITY BEAMS IN NB COORDS.
27 VYBEAMNB       EQUALS  VZBEAMNB +6   # I(6)
28 VXBEAMNB       EQUALS  VYBEAMNB +6   # I(6) PRESERVE Z,Y,X ORDER
29
30 LRVTIME        =       VXBEAMNB +6   # B(2) LR
31 LRXCDU         =       LRVTIME +2    # B(1) LR
32 LRYCDU         =       LRXCDU +1     # B(1) LR
33 LRZCDU         =       LRYCDU +1     # B(1) LR
34 PIPTM          =       LRZCDU +1     # B(3) LR
35
36 # P32-P35, P72-P75 STORAGE.      (40D)
37
38 T1TOT2         ERASE   +1            # (2)  TIME FROM CSI TO CDH
39 T2TOT3         ERASE   +1            # (2)
40 ELEV           ERASE   +1            # (2)
41 UP1            ERASE   +5            # (6)
42 DELVEET1       ERASE   +5            # I(6)  DV CSI IN REF
43 DELVEET2       ERASE   +5            # I(6)  DV CSH IN REF
44 RACT1          ERASE   +5            # (6)   POS VEC OF ACTIVE AT CSI TIME
45 RACT2          ERASE   +5            # (6)   POS VEC OF ACTIVE AT CDH TIME
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1  RTSR1/MU      ERASE  +1      # (2)  SQ ROOT 1/MU STORAGE
2  RTMU          ERASE  +1      # (2)  MU STORAGE
3
4
5  # (THE FOLLOWING ERASABLES OVERLAY PORTIONS OF THE PREVIOUS SECTION)
6
7  +MGA          EQUALS  T1TOT2   # (2)  S-S + MID GIM ANGL TO DELVEET3
8
9  UNRM          EQUALS  UP1      # I(6)  S-S
10
11  DVLOS         EQUALS  RACT1     # I(6)  S-S DELTA VELOCITY, LOS COORD-DISPLAY
12  ULOS          EQUALS  RACT2     # I(6)  S-S UNIT LINE OF SIGHT VECTOR
13
14  NOMTPI        EQUALS  RTSR1/MU  # (2)  S-S NOMINAL TPI-TIME FOR RECYCLE
15
16  # SOME P30 STORAGE.                (4D)
17
18  HAPO          EQUALS  RTSR1/MU  # I(2)
19  HPER          EQUALS  HAPO +2    # I(2)
20
21
22  # SOME P38-P39,P78-79 STORAGE      #      (6D)
23
24  DELTAR        EQUALS  DVLOS     # I(2)
25  DELTTIME      EQUALS  DELTAR +2  # I(2) TIME REPRESENTATION OF DELTAR
26  TARGTIME      EQUALS  DELTTIME +2 # I(2) TINT MINUS DELTTIME
27
28  TINTSOI       EQUALS  DELTAR    # I(2) TIME OF INTERCEPT FOR SOI PHASE
29
30  # THE FOLLOWING ARE ERASABLE LOADS DURING A PERFORMANCE TEST.
31
32  TRANSM1       =      WRENDPOS   # E4,1400
33  ALFDK         =      TRANSM1 +18D
34
35  # ***** THE FOLLOWING SECTIONS OVERLAY V83 AND DISPLAY STORAGE *****
36
37  # V47 (R47) AGS INITIALIZATION PROGRAM STORAGE. (OVERLAYS V83) (14D)
38
39  AGSBUFF       EQUALS  RANGE     # B(14D)
40  AGSBUFFE      EQUALS  AGSBUFF +13D # ENDMARK
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

R36 OUT-OF-PLANE RENDEZVOUS DISPLAY STORAGE. (OVERLAYS V83) (12D)

RPASS36 EQUALS RONE # I(6) S-S
UNP36 EQUALS RPASS36 +6 # I(6) S-S

S-BAND ANTENNA GIMBAL ANGLES. DISPLAYED BY R05 (V64). (OVERLAYS V83) (10D)
(OPERATES DURING P00 ONLY)

ALPHASB EQUALS RANGE # B(2)DSP NOUN 51. PITCH ANGLE.
BETASB EQUALS ALPHASB +2 # B(2)DSP NOUN 51. YAW ANGLE.
RLM EQUALS BETASB +2 # I(6)S S/C POSITION VECTOR.

**** USED IN S-BAND ANTENNA FOR LM ****

YAWANG EQUALS BETASB
PITCHANG EQUALS ALPHASB

NOUN 56 DATA -- COMPUTED AND DISPLAYED BY VERB 85.

RR-AZ EQUALS PITCHANG # I(2) ANGLE BETWEEN LOS AND X-Z PLANE
RR-ELEV EQUALS RR-AZ +2 # I(2) ANGLE BETWEEN LOS AND Y-Z PLANE

R04 (V62) RADAR TEST STORAGE.
R04 IS RESTRICTED TO P00.

RSTACK EQUALS RANGE # B(8) BUFFER FOR R04 NOUNS.

INITVEL STORAGE. ALSO USED BY P31, P34, P35, P74, P75, P10, P11, MIDGIM, S40.1 AND S40.9. (18D)

(POSSIBLY RINIT & VINIT CAN OVERLAY DELVEET1 & 2 ABOVE)

RINIT ERASE +5 # I(6) ACTIVE VEHICLE POSITION
VINIT ERASE +5 # I(6) ACTIVE VEHICLE VELOCITY
VIPRIME ERASE +5 # I(6) NEW VEL REQUIRED AT INITIAL RADIUS.

VARIOUS DISPLAY REGISTERS. BALLANGS (3D)

FDAIX ERASE # I(1)
FDAIY ERASE # I(1)
FDAIZ ERASE # I(1)

P34-P35 STORAGE. DOWNLINKED. (2D)

DELVTPF ERASE +1 # I(2) DELTA V FOR TPF

SOME R04(V63)-R77 RADAR TEST STORAGE. (6D)


```
1 RTSTDEX      ERASE      # (1)
2 RTSTMAX      ERASE      # (1)
3 RTSTBASE     ERASE      # (1)
4 RTSTLOC      ERASE      # (1)
5 RSTKLOC      =          RTSTLOC
6 RSAMPDT      ERASE      # (1)
7 RFAILCNT     ERASE      # (1)
8
9 # LPS20.1 STORAGE.                (12D)
10
11 LMPOS        EQUALS RTSTDEX      # I(6)TMP  STORAGE FOR LM POS. VECTOR.
12 LMVEL        EQUALS LMPOS +6     # I(6)TMP  STORAGE FOR LM VEL. VECTOR.
13
14 # INITVEL STORAGE.  ALSU USED BY P31,34,35,74,75,S40.1 AND DOWNLINKED.  (6D)
15
16 DELVEET3     EQUALS  LMVEL +6    # I(6) DELTA V IN INERTIAL COORDINATES
17
18 END-E4        EQUALS              # FIRST UNUSED LOCATION IN E4
19
20 # SECOND DPS GUIDANCE (LUNAR LANDING) (OVERLAY P32-35, INITVEL)      (14D)
21
22 VHORIZ        EQUALS  PIPTM +3   # I(2) DISPLAY
23 ANGTERM        EQUALS  VHORIZ +2 # I(6) GUIDANCE
24 HBEAMNB        EQUALS  ANGTERM +6 # I(6) LANDING RADAR
25
26 # R12 DOWNLINK QUANTITIES                (5D)
27
28 LRXCDUDL      EQUALS  /LAND/ +2  # B(1) LANDING RADAR DOWNLINK
29 LRYCDUDL      EQUALS  LRXCDUDL +1 # B(1) LANDING RADAR DOWNLINK
30 LRZCDUDL      EQUALS  LRYCDUDL +1 # B(1) LANDING RADAR DOWNLINK
31 LRVTIMDL      EQUALS  LRZCDUDL +1 # B(2) LANDING RADAR DOWNLINK
32
33 # ASCENT GUIDANCE FOR LUNAR LANDING      (54D)
34
35 AT            EQUALS  PIPTM +3   # I(2)TMP ENGINE DATA -- THRUST ACC*2(9)
36 VE            EQUALS  AT +2      # I(2)TMP EXHAUST VELOCITY * 2(7)M/CS.
37 TTO           EQUALS  VE +2      # I(2)TMP TAILOFF TIME * 2(17)CS.
38 TBUP          EQUALS  TTO +2     # I(2)TMP (M/MDOT) * 2(17)CS.
39 RDOTD         EQUALS  TBUP +2    # I(2)TMP TARGET VELOCITY COMPONENTS
40 YDOTD         EQUALS  RDOTD +2   # I(2)TMP SCALING IS 2(7)M/CS.
41 ZDOTD         EQUALS  YDOTD +2   # I(2)TMP
42
43 /R/MAG        EQUALS  ZDOTD +2   # I(2)TMP
44 LAXIS         EQUALS  /R/MAG +2  # I(6)TMP
```

```
1  ZAXIS1      =      UHZP
2  RDOT        =      HDOTDISP
3
4  YDOT        =      LAXIS +6      # I(2)TMP VEL. NORMAL TO REF. PLANE*2(-7)
5  ZDOT        EQUALS  YDOT +2      # I(2)TMP DOWN RANGE VEL * 2(-7)
6  GEFF        EQUALS  ZDOT +2      # I(2)TMP EFFECTIVE GRAVITY
```

```
7
8  # THESE TWO GROUPS OF ASCENT GUIDANCE ARE SPLIT BY THE ASCENT-DESCENT SERVICER SECTION FOLLOWING THIS SECTION
9
```

```
10 Y           EQUALS  /LAND/ +2      # I(2)TMP OUT-OF-PLANE DIST *2(24)M
11 DRDOT       EQUALS  Y +2           # I(2)TMP RDOTD - RDOT
12 DYDOT       EQUALS  DRDOT +2       # I(2)TMP YDOTD - YDOT
13 DZDOT       EQUALS  DYDOT +2       # I(2)TMP ZDOTD - ZDOT
14 PCONS       EQUALS  DZDOT +2       # I(2)TMP CONSTANT IN ATR EQUATION
15 YCONS       EQUALS  PCONS +2       # I(2)TMP CONSTANT IN ATY EQUATION
16 PRATE       EQUALS  YCONS +2       # I(2)TMP RATE COEFF. IN ATR EQUATION
17 YRATE       EQUALS  PRATE +2       # I(2)TMP RATE COEFF. IN ATY EQUATION
18 ATY         EQUALS  YRATE +2       # I(2)TMP OUT-OF-PLANE THRUST COMP. *2(9)
19 ATR         EQUALS  ATY +2         # I(2)TMP RADIAL THRUST COMP. * 2(9)
20 ATP         EQUALS  ATR +2         # I(2)TMP DOWN-RANGE THRUST COMP
21 YAW         EQUALS  ATP +2         # I(2)TMP
22 PITCH       EQUALS  YAW +2         # I(2)TMP
```

```
23
24 # SERVICER FOR LUNAR ASCENT AND DESCENT      (14D)
```

```
25
26 G(CSM)       EQUALS  GEFF +2      # I(6) FOR UPDATE OF COMMAND MODULE STATE
27 R(CSM)       EQUALS  R-OTHER      # VECTORS BY LEM: ANALOGS OF GDT/2,
28 V(CSM)       EQUALS  V-OTHER      # R, AND V, RESPECTIVELY OF THE CSM
29 WM          EQUALS  G(CSM) +6     # I(6)TMP -- LUNAR ROTATION VECTOR (SM)
30 /LAND/       EQUALS  WM +6        # B(2) LUNAR RADIUS AT LANDING SITE
```

EBANK-5 ASSIGNMENTS

SETLOC 2400

W-MATRIX. ESSENTIALLY UNSHARABLE. (162D)

W ERASE +161D
ENDW EQUALS W +162D

***** OVERLAY NUMBER 1 IN EBANK 5 *****

W-MATRIX PADLOADS (124D)

TLAND	EQUALS	W	# I(2)	NOMINAL TIME OF LANDING
RBRFG	EQUALS	TLAND +2	# I(6)	BRAKING
VBRFG	EQUALS	RBRFG +6	# I(6)	PHASE
ABRFG	EQUALS	VBRFG +6	# I(6)	TARGET
VBRFG*	EQUALS	ABRFG +6	# I(2)	PARAMETERS:
ABRFG*	EQUALS	VBRFG* +2	# I(2)	HIGH
JBRFG*	EQUALS	ABRFG* +2	# I(2)	GATE
GAINBRAK	EQUALS	JBRFG* +2	# B(2)	
TCGFBRAK	EQUALS	GAINBRAK +2	# B(1)	
TCGIBRAK	EQUALS	TCGFBRAK +1	# B(1)	
RAPFG	EQUALS	TCGIBRAK +1	# I(6)	APPROACH
VAPFG	EQUALS	RAPFG +6	# I(6)	PHASE
AAPFG	EQUALS	VAPFG +6	# I(6)	TARGET
VAPFG*	EQUALS	AAPFG +6	# I(2)	PARAMETERS:
AAPFG*	EQUALS	VAPFG* +2	# I(2)	LOW
JAPFG*	EQUALS	AAPFG* +2	# I(2)	GATE
GAINAPPR	EQUALS	JAPFG* +2	# B(2)	
TCGFAPPR	EQUALS	GAINAPPR +2	# B(1)	
TCGIAPPR	EQUALS	TCGFAPPR +1	# B(1)	
VIGN	EQUALS	TCGIAPPR +1	# I(2)	DESIRED SPEED FOR IGNITION
RIGNX	EQUALS	VIGN +2	# I(2)	DESIRED 'ALTITUDE' FOR IGNITION
RIGNZ	EQUALS	RIGNX +2	# I(2)	DESIRED GROUND RANGE FOR IGNITION
KIGNX/B4	EQUALS	RIGNZ +2	# I(2)	
KIGNY/B8	EQUALS	KIGNX/B4 +2	# I(2)	
KIGNV/B4	EQUALS	KIGNY/B8 +2	# I(2)	
LOWCRIT	EQUALS	KIGNV/B4 +2	# B(1)	(HIGHCRIT MUST FOLLOW LOWCRIT)
HIGHCRIT	EQUALS	LOWCRIT +1	# B(1)	
V2FG	EQUALS	HIGHCRIT +1	# I(6)	DESIRED VELOCITY FOR P65.
TAUVERT	EQUALS	V2FG +6	# I(2)	TIME CONSTANT FOR P65 VEL. NULLING.
DELQFIX	EQUALS	TAUVERT +2	# I(2)	LR ALTITUDE DATA REASONABLE PARM.
LRALPHA	EQUALS	DELQFIX +2	# B(1)	POS1 X ROTATION * MUST *

```
1  LRBETA1      EQUALS  LRALPHA +1      # B(1) POS1 Y ROTATION      * BE *
2  LRALPHA2     EQUALS  LRBETA1 +1      # B(1) POS2 X ROTATION      * IN *
3
4  LRBETA2      EQUALS  LRALPHA2 +1     # B(1) POS2 Y ROTATION      * ORDER*
5  LRVMAX       EQUALS  LRBETA2 +1     # B(1) LR VEL WEIGHTING FUNCTIONS
6  LRVF         EQUALS  LRVMAX +1      # B(1) LR VEL WEIGHTING FUNCTIONS
7
8  LRWVZ        EQUALS  LRVF +1         # B(1) LR VEL WEIGHTING FUNCTIONS
9  LRWVY        EQUALS  LRWVZ +1       # B(1) LR VEL WEIGHTING FUNCTIONS
10 LRWVX         EQUALS  LRWVY +1       # B(1) LR VEL WEIGHTING FUNCTIONS
11
12 LRWVFZ        EQUALS  LRWVX +1       # B(1) LR VEL WEIGHTING FUNCTIONS
13 LRWVFY        EQUALS  LRWVFZ +1     # B(1) LR VEL WEIGHTING FUNCTIONS
14 LRWVFX        EQUALS  LRWVFY +1     # B(1) LR VEL WEIGHTING FUNCTIONS
15 LRWVFF        EQUALS  LRWVFX +1     # B(1) LR VEL WEIGHTING FUNCTIONS
16
17 ABVEL*        EQUALS  BUF            # B(1) LR TEMP
18 VSELECT*      EQUALS  BUF +1         # B(1) LR TEMP
19
20 RODSCALE      EQUALS  LRWVFF +1      # I(1) CLICK SCALE FACTOR FOR ROD
21 TAUROD        EQUALS  RODSCALE +1    # I(2) TIME CONSTANT FOR R.O.D.
22 LAG/TAU       EQUALS  TAUROD +2      # I(2) LAG TIME DIVIDED BY TAUROD (P66)
23 MINFORCE      EQUALS  LAG/TAU +2     # I(2) MINIMUM FORCE P66 WILL COMMAND
24 MAXFORCE      EQUALS  MINFORCE +2    # I(2) MAXIMUM FORCE P66 WILL COMMAND.
25 ABTCOF        EQUALS  MAXFORCE +2    # I(16) COEFFICIENTS FOR ABORT TFI POLYS.
26 VMIN          EQUALS  ABTCOF +16D    # I(2) MINIMUM VELOCITY FOR ABORT INJ.
27 YLIM          EQUALS  VMIN +2        # I(2) MAXIMUM CROSS-RANGE DIST. IN ABORTS
28 ABTRDOT       EQUALS  YLIM +2        # I(2) DESIRED RADIAL VEL. FOR ABORTS.
29 COSTHET1      EQUALS  ABTRDOT +2     # I(2) COS CONE 1 ANGLE FOR ABORTS.
30 COSTHET2      EQUALS  COSTHET1 +2    # I(2) COS OF CONE 2 ANGLE FOR ABORTS.
31
32 # SOME VARIABLES FOR SECOND DPS GUIDANCE.      (34D)
33
34 CG            EQUALS  COSTHET2 +2    # I(18D) GUIDANCE
35 RANGEDSP      EQUALS  CG +18D        # B(2) DISPLAY
36 OUTOFPLN      EQUALS  RANGEDSP +2    # B(2) DISPLAY
37 R6OVSAVE      EQUALS  OUTOFPLN +2    # I(6) TMP SAVES VALUE OF POINTVSM THRU R51
38 RGU           EQUALS  R6OVSAVE +6    # I(6) UNSHARED FOR DOWNLINK
39 VBIAS         EQUALS  R6OVSAVE       # I(6) PIPA BIAS EQUIV. VELOCITY VECTOR.
40 L*WCR*T       =      BUF
41 H*GHCRT       =      BUF +1
42
43 # ALIGNMENT/SYSTEST/CALCSMSC COMMON STORAGE      (36D)
44
45 XSM           EQUALS  ENDW            # B(6)
46 YSM           EQUALS  XSM +6          # B(6)
47 ZSM           EQUALS  YSM +6          # B(6)
48
49 XDC           EQUALS  ZSM +6          # B(6)
50 YDC           EQUALS  XDC +6          # B(6)
51 ZDC           EQUALS  YDC +6          # B(6)
```

```
1 XNB      =      XDC
2 YNB      =      YDC
3 ZNB      =      ZDC
```

OVERLAYS WITHIN ALIGNMENT/SYSTEST/CALCSMSC COMMON STORAGE (4D)

```
8 -COSB      EQUALS  XSM +2      # (2)TMP
9 SINB      EQUALS  -COSB +2     # (2)TMP
```

MORE OVERLAYS TO ALIGNMENT/SYSTEST (THESE ARE P52) (6D)

```
13 LANDLAT    EQUALS  STARAD      # (2) LATTITUDE, LONGITUDE
14 LANDLONG   EQUALS  LANDLAT +2   # (2)  AND ALTITUDE
15 LANDALT    EQUALS  LANDLONG +2  # (2)  OF LANDING SITE
```

ALIGNMENT/SYSTEST COMMON STORAGE. (31D)

```
19 STARAD     EQUALS  ZDC +6      # I(18D)TMP
20 STAR       EQUALS  STARAD +18D  # I(6)
21 GCTR       EQUALS  STAR +6      # B(1)
22 OGC        EQUALS  GCTR +1      # I(2)
23 IGC        EQUALS  OGC +2       # I(2)
24 MGC        EQUALS  IGC +2       # I(2)
```

P57 ALIGNMENT (OVERLAY OF ALIGNMENT/SYSTEST COMMON STORAGE) (12D)

```
28 GACC       =      STARAD      # (6) SS
29 GOUT       =      STARAD +6    # (6) SS
```

OVERLAYS WITHIN ALIGNMENT/SYSTEST COMMON STORAGE (24D)

```
33 VEARTH     EQUALS  STARAD      # (6)TMP
34 VSUN       EQUALS  VEARTH +6    # (6)TMP
35 VMOON      EQUALS  VSUN +6      # (6)TMP
36 SAX        EQUALS  VMOON +6     # (6)TMP
```

P50'S, R50'S Q STORES (2D)

```
40 QMIN       EQUALS  MGC +2      # B(1)TMP
41 QMAJ       EQUALS  QMIN +1     # B(1)TMP
```

**** USED IN P50S **** (SCATTERED OVERLAYS)

```
45 XSCI       EQUALS  STARAD
46 YSCI       EQUALS  XSCI +6
```

ERASABLE_ASSIGNMENTS

1						1
2	ZSCI	EQUALS	YSCI			2
3	CULTRIX	EQUALS	VEARTH	# VEARTH, VSUN, VMOON		3
4	VEC1	EQUALS	STARAD +12D			4
5	VEC2	EQUALS	STAR			5
6						6
7	# ALIGNMENT STORAGE. (23D)					7
8						8
9	OGCT	EQUALS	QMAJ +1	# I(6)		9
10	BESTI	EQUALS	OGCT +6	# I(1)		10
11	BESTJ	EQUALS	BESTI +1			11
12	STARIND	EQUALS	BESTJ +1			12
13	# RETAIN THE ORDER OF STARS AV1 TO STARS AV2 +5 FOR DOWNLINK PURPOSES					13
14	STARS AV1	EQUALS	STARIND +1	# I(6)		14
15	STARS AV2	EQUALS	STARS AV1 +6	# I(6)		15
16	TALIGN	EQUALS	STARS AV2 +6	# B(2) TIME OF IMU ALIGNMENT (DOWNLINKED)		16
17						17
18	# P32-35 + SERVICER					18
19						19
20	RTX1	EQUALS	TALIGN +2	# I(1) X1	-2 EARTH, -10 MOON	20
21	RTX2	EQUALS	RTX1 +1	# I(1) X2	0 EARTH, 2 MOON	21
22						22
23	ZPRIME	=	22D			23
24	PDA	=	22D			24
25	COSTH	=	16D			25
26	SINTH	=	18D			26
27	THETA	=	20D			27
28	STARM	=	32D			28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

***** OVERLAY NUMBER 2 IN EBANK 5 *****

CONICS ROUTINE STORAGE. (85D)

DELX	EQUALS	ENDW	# I(2)TMP
DELT	EQUALS	DELX +2	# I(2)TMP
URRECT	EQUALS	DELT +2	# I(6)TMP
RCNORM	EQUALS	34D	# I(2)TMP
XPREV	EQUALS	XKEP	# I(2)TMP
R1VEC	EQUALS	URRECT +6	# I(6)TMP
R2VEC	EQUALS	R1VEC +6	# I(6)TMP
TDESIRED	EQUALS	R2VEC +6	# I(2)TMP
GEOMSGN	EQUALS	TDESIRED +2	# I(1)TMP
UN	EQUALS	GEOMSGN +1	# I(6)TMP
VTARGETAG	EQUALS	UN +6	# I(1)TMP
VTARGET	EQUALS	VTARGETAG +1	# I(6)TMP
RTNLAMB	EQUALS	VTARGET +6	# I(1)TMP
U2	EQUALS	RTNLAMB +1	# I(6)TMP
MAGVEC2	EQUALS	U2 +6	# I(2)TMP
UR1	EQUALS	MAGVEC2 +2	# I(6)TMP
SNTH	EQUALS	UR1 +6	# I(2)TMP
CSTH	EQUALS	SNTH +2	# I(2)TMP
1-CSTH	EQUALS	CSTH +2	# I(2)TMP
CSTH-RHO	EQUALS	1-CSTH +2	# I(2)TMP
P	EQUALS	CSTH-RHO +2	# I(2)TMP
R1A	EQUALS	P +2	# I(2)TMP
RVEC	EQUALS	R1VEC	# I(6)TMP
VVEC	EQUALS	R1A +2	# I(6)TMP
RTNTT	EQUALS	RTNLAMB	# I(1)TMP
ECC	EQUALS	VVEC +6	# I(2)TMP
RTNTR	EQUALS	RTNLAMB	# I(1)TMP
RTNAPSE	EQUALS	RTNLAMB	# I(1)TMP
R2	EQUALS	MAGVEC2	# I(2)TMP
RTNPRM	EQUALS	ECC +2	# I(1)TMP
SGNRDOT	EQUALS	RTNPRM +1	# I(1)TMP
RDESIRED	EQUALS	SGNRDOT +1	# I(2)TMP
DELDEP	EQUALS	RDESIRED +2	# I(2)TMP
DEPREV	EQUALS	DELDEP +2	# I(2)TMP
TERRLAMB	EQUALS	DELDEP	# I(2)TMP
TPREV	EQUALS	DEPREV	# I(2)TMP
EPSILONL	EQUALS	DEPREV +2	# I(2)TMP
COGA	EQUALS	EPSILONL +2	# I(2) COTAN OF INITIAL FLIGHT PATH ANGLE.
INDEP	EQUALS	COGA	# USED BY SUBROUTINE `ITERATOR'.

***** OVERLAY NUMBER 3 IN EBANK 5 *****

INCORP STORAGE. (18D)

ZI EQUALS ENDW # I(18D)TMP

INCORP/L SR22.3 STORAGE. (21D)

DELTAX EQUALS ZI +18D # I(18)

VARIANCE EQUALS DELTAX +18D # I(3)

MEASUREMENT INCORPORATION -R22- STORAGE. (49D)

GRP2SVQ EQUALS VARIANCE +3 # I(1)TMP QSAVE FOR RESTARTS

OMEGAM1 EQUALS GRP2SVQ +1 # I(6)

OMEGAM2 EQUALS OMEGAM1 +6 # I(6)

OMEGAM3 EQUALS OMEGAM2 +6 # I(6)

HOLDW EQUALS OMEGAM3 +6 # I(18)

TDPOS EQUALS HOLDW +18D # I(6)

TDVEL EQUALS TDPOS +6 # I(6)

TRIPA EQUALS DELTAX # I(3)TMP

TEMPVAR EQUALS TRIPA +3 # I(3)TMP

INCORPORATION/INTEGRATION Q STORAGE. (1D)

EGRESS EQUALS TDVEL +6 # I(1)

P30/P31 STORAGE. (1D) AND ONE OVERLAY

P30EXIT EQUALS EGRESS +1 # B(1)TMP

ORIGIN EQUALS P30EXIT # I(1)TMP INTEX DURING INITVEL.

SYSTEM TEST ERASABLES. CAN OVERLAY W MATRIX. (127D)

***** OVERLAY NUMBER 0 IN EBANK 5 *****

AZIMUTH EQUALS W # 2

LATITUDE EQUALS AZIMUTH +2 # 2

ERVECTOR EQUALS LATITUDE +2 # 6

LENGTHOT EQUALS ERVECTOR +6 # 1

LOSVEC EQUALS LENGTHOT +1 # 6

NDXCTR EQUALS LOSVEC +1 # 1

PIPINDEX EQUALS NDXCTR +1 # 1

POSITON EQUALS PIPINDEX +1 # 1

QPLACE EQUALS POSITON +1 # 1

QPLACES EQUALS QPLACE +1 # 1

SOUTHDR EQUALS QPLACES +1 # 7

TEMPTIME EQUALS SOUTHDR +7 # 2

TMARK EQUALS TEMPTIME +2 # 2

GENPL EQUALS TMARK +2

CDUTIMEI = GENPL

CDUTIMEF = GENPL +2

CDUDANG = GENPL +4

CDUREADF = GENPL +5

CDUREADI = GENPL +6

CDULIMIT = GENPL +7

TEMPADD = GENPL +4

TEMP = GENPL +5

NOBITS = GENPL +6

CHAN = GENPL +7

LOS1 = GENPL +8D

LOS2 = GENPL +14D

CALCDIR EQUALS GENPL +20D

CDUFLAG EQUALS GENPL +21D

GYTOBETQ EQUALS GENPL +22D

OPTNREG EQUALS GENPL +23D

SAVE EQUALS GENPL +24D # THREE ONSEC LOC

SFCONST1 EQUALS GENPL +27D

TIMER EQUALS GENPL +28D

DATAPL EQUALS GENPL +30D

RDSP EQUALS GENPL # FIX LATER POSSIBLY KEEP1

MASKREG EQUALS GENPL +64D

CDUNDX EQUALS GENPL +66D

RESULTCT EQUALS GENPL +67D

COUNTPL EQUALS GENPL +70D

CDUANG EQUALS GENPL +71D

AINLA	=	GENPL	# 110 DEC OR 156 OCT LOCATIONS
WANGO	EQUALS	AINLA	# VERT ERATE
WANGI	EQUALS	AINLA +2D	# HORIZONTAL ERATE
WANGT	EQUALS	AINLA +4D	# T
TORQNDX	=	WANGT	
DRIFTT	EQUALS	AINLA +6D	
ALXIS	EQUALS	AINLA +8D	
CMPX1	EQUALS	AINLA +9D	# IND
ALK	EQUALS	AINLA +10D	# GAINS
VLAUNS	EQUALS	AINLA +22D	
WPLATO	EQUALS	AINLA +24D	
INTY	EQUALS	AINLA +28D	# SOUTH IP INTE
ANGZ	EQUALS	AINLA +30D	# EAST A IS
INTZ	EQUALS	AINLA +32D	# EAST P P I
ANGY	EQUALS	AINLA +34D	# SOUTH
ANGX	EQUALS	AINLA +36D	# VE
DRIFTO	EQUALS	AINLA +38D	# VERT
DRIFTI	EQUALS	AINLA +40D	# SOU
VLAUN	EQUALS	AINLA +44D	
ACCWD	EQUALS	AINLA +46D	
POSNV	EQUALS	AINLA +52D	
DPIPAY	EQUALS	AINLA +54D	# SOUTH
DPIPAZ	EQUALS	AINLA +58D	# NORTH IP INCREMENT
ALTIM	EQUALS	AINLA +60D	
ALTIMS	EQUALS	AINLA +61D	# INDEX
ALDK	EQUALS	AINLA +62D	# TIME ONSTAN
DELM	EQUALS	AINLA +76D	
WPLATI	EQUALS	AINLA +84D	
GEOCOMPS	EQUALS	AINLA +86D	
ERCOMP	EQUALS	AINLA +87D	
ZERONDX	EQUALS	AINLA +93D	
THETAN	=	ALK +4	
FILDELV	EQUALS	THETAN +6	# AGS ALIGNMENT STORAGE
INTVEC	EQUALS	FILDELV +2	
ISECXT	=	AINLA +94D	
ASECXT	=	AINLA +95D	
PERFDLAY	EQUALS	AINLA +96D	# B(2) DELAY TIME BEF. START DRIFT MEASURE
OVFLOWCK	EQUALS	AINLA +98D	# (1) SET MEANS OVERFLOW IN IMU PERF TEST
END-E5	EQUALS	STARSAV2 +6	# *** FIRST FREE LOCATION IN E5 ***

EBANK-6 ASSIGNMENTS.

SETLOC 3000

DAP PAD-LOADED DATA. (10D)

ALL OF THE FOLLOWING EXCEPT PITTIME AND ROLLTIME ARE INITIALIZED IN FRESH START TO PERMIT IMMEDIATE USE OF DAP.

HIASCENT	ERASE	# (1) MASS AFTER STAGING, SCALE AT B16 KG.
ROLLTIME	ERASE	# (1) TIME TO TRIM Z GIMBAL IN R03, CSEC.
PITTIME	ERASE	# (1) TIME TO TRIM Y GIMBAL IN R03, CSEC.
DKTRAP	ERASE	# (1) DAP STATE (POSSIBLE 77001
DKOMEGAN	ERASE	# (1) ESTIMATOR PARA- (VALUES 00012
DKKAOSN	ERASE	# (1) METERS FOR THE 00074
LMTRAP	ERASE	# (1) DOCKED AND 77001
LMOMEGAN	ERASE	# (1) LEM-ALONE CASES 00000
LMKAOSN	ERASE	# (1) RESPECTIVELY 00074
DKDB	ERASE	# (1) WIDTH OF DEADBAND FOR DOCKED RCS
		# AUTOPILOT (DB=1.4DEG IN FRESH START
		# DEADBAND = PI/DKDB RAD.

PADLOADS FOR INITIALIZATION OF DAP BIAS ACCELERATION (AT P12 IGNITION) (2D)

IGNAOSQ	ERASE	# B(1)PL
IGNAOSR	ERASE	# B(1)PL

AXIS TRANSFORMATION MATRIX -- GIMBAL TO PILOT AXES: (5D)

M11	ERASE	# SCALED AT 1
M21	ERASE	# SCALED AT 1
M31	ERASE	
M22	ERASE	# SCALED AT 1.
M32	ERASE	# SCALED AT 1.

ANGLE MEASUREMENTS

OMEGAP	ERASE	+4	# BODY-AXIS ROT. RATES SCALED AT PI/4 AND
OMEGAQ	EQUALS	OMEGAP +1	# BODY-AXIS ACCELERATIONS SCALED AT PI/8.
OMEGAR	EQUALS	OMEGAP +2	
# RETAIN THE ORDER OF ALPHAQ AND ALPHAR FOR DOWNLINK PURPOSES.			
ALPHAQ	EQUALS	OMEGAP +3	
ALPHAR	EQUALS	OMEGAP +4	
OMEGAU	ERASE	+1	
OMEGAV	=	OMEGAU +1	
TRAPEDP	ERASE	+5	
TRAPEDQ	=	TRAPEDP +1	
TRAPEDR	=	TRAPEDP +2	
NPTRAPS	=	TRAPEDP +3	

```
1  NQTRAPS      =      TRAPEDP +4
2  NRTRAPS      =      TRAPEDP +5
3
4  EDOTP         =      EDOT
5  EDOTQ         ERASE   +1
6  EDOTR         =      EDOTQ +1      # MANY SHAREING NAMES
7  QRATEDIF      EQUALS  EDOTQ      # ALTERNATIVE NAMES:
8  RRATEDIF      EQUALS  EDOTR      # DELETE WHEN NO. OF REFERENCES = 0
9
10 URATEDIF      EQUALS  OMEGAU
11 VRATEDIF      EQUALS  OMEGAV
12 OLDXFORP      ERASE   +2      # STORED CDU READINGS FOR STATE
13 OLDYFORP      EQUALS  OLDXFORP +1 # DERIVATIONS: SCALED AT PI RADIANS (2'S)
14 OLDZFORQ      EQUALS  OLDXFORP +2
15
16 # RATE-COMMAND AND MINIMUM IMPULSE MODES
17
18 CH31TEMP      ERASE
19 STIKSENS      ERASE
20 TCP           ERASE
21 DXERROR       ERASE   +5
22 DYERROR       EQUALS  DXERROR +2
23 DZERROR       EQUALS  DXERROR +4
24 PLAST         ERASE
25 QLAST         ERASE
26 RLAST         ERASE
27 TCQR          ERASE
28
29 # OTHER VARIABLES                      (5D)
30
31 OLDPMIN        ERASE      # THESE THREE USED IN MIN IMPUSE MODE
32 OLDQRMIN       ERASE
33 TEMP31         EQUALS  DAPTEMP1
34
35 SAVEHAND       ERASE   +1
36 PERROR         ERASE
37 QERROR         EQUALS  DYERROR
38 RERROR         EQUALS  DZERROR
39
40 # JET STATE CHANGE VARIABLES -- TIME (TOFJTCHG), JET BITS WRITTEN NOW   (10D)
41 # (JTSONNOW), AND JET BITS WRITTEN AT T6 RUPT (JTSATCHG).
42
43 NXT6ADR        ERASE
44 T6NEXT         ERASE   +1
45 T6FURTHA       ERASE   +1
46 NEXTP          ERASE   +2
47 NEXTU          =      NEXTP +1
48 NEXTV          =      NEXTP +2
49 -2JETLIM       ERASE   +1      # RATE COMMAND 4-JET RATE DIFFERENCE LIMIT
50 -RATEDB        EQUALS  -2JETLIM +1 # AND RATE DEADBAND FOR ASCENT OR DESCENT
51
52 TARGETDB       EQUALS  -RATEDB      # MAN. CONTROL TARGET DB COMPLEMENT.
53
54 # ***Q,R AXIS ERASABLES ***          (3)
```

1412THE

TRIM GIMBAL CONTROL LAW ERASABLES:

(11D)

GTSTEMPS	EQUALS	DAPTEMP1	# GTS IS PART OF THE JASK.
SHFTFLAG	EQUALS	GTSTEMPS +2	# COUNT BITS FOR GTSQRT SHIFTING.
ININDEX	EQUALS	GTSTEMPS +5	# INDEX FOR SHIFT LOOP IN GTSQRT.
SAVESR	EQUALS	AXISCTR	# CANNOT BE A DAPTEMP -- GTS USES THEM ALL.
SCRATCH	EQUALS	GTSTEMPS +7	# ROOTCYCL ERASABLE.
HALFARG	EQUALS	GTSTEMPS +8D	# ROOTCYCL ERASABLE.
K2THETA	EQUALS	GTSTEMPS	# D.P., K*ERROR, NEGUSUM
KCENTRAL	EQUALS	GTSTEMPS +2	# S.P., K FROM KQ OR KRDP, AT PI/2(8)
K2CNTRAL	EQUALS	GTSTEMPS +3	# D.P., GTS SCRATCH CELLS.
WCENTRAL	EQUALS	GTSTEMPS +4	# S.P., OMEGA, AT PI/4 RAD/SEC
ACENTRAL	EQUALS	GTSTEMPS +5	# S.P., ALPHA, AT PI/4 RAD/SEC(2)
DEL	EQUALS	GTSTEMPS +6	# S.P., SGN FUNCTION VALUE.
A2CNTRAL	EQUALS	GTSTEMPS +7	# D.P., GTS SCRATCH CELLS.
QRCNTR	EQUALS	GTSTEMPS +9D	# S.P., INDEX FOR GTS LOOP THROUGH Q,R AXES
FUNCTION	EQUALS	GTSTEMPS +10D	# D.P., ARGUMENT FOR GRSQRT, SCRATCH FOR GTS.

NEGUQ	ERASE	+2	# NEGATIVE OF Q-AXIS GIMBAL DRIVE.
	EQUALS	NEGUQ +1	# DEFINED AND USED ELSEWHERE.

NEGUR	EQUALS	NEGUQ +2	# NEGATIVE OF R-AXIS GIMBAL DRIVE.
-------	--------	----------	------------------------------------

KQ	ERASE	+2	# S.P., JERK TERM FOR GTS, AT PI/2(8)
----	-------	----	---------------------------------------

AXISCTR	EQUALS	KQ +1	
KRDAP	EQUALS	KQ +2	# .3 ACCDOTR SCALED AT PI/2(8)

ACCDOTQ	ERASE	+3	# Q-JERK SCALED AT PI/2(7) UNSIGNED
---------	-------	----	-------------------------------------

QACCDOT	EQUALS	ACCDOTQ +1	# Q-JERK SCALED AT PI/2(7) SIGNED
---------	--------	------------	-----------------------------------

ACCDOTR	EQUALS	ACCDOTQ +2	# R-JERK SCALED AT PI/2(7) UNSIGNED
---------	--------	------------	-------------------------------------

RACCDOT	EQUALS	ACCDOTQ +3	# R-JERK SCALED AT PI/2(7) SIGNED
---------	--------	------------	-----------------------------------

QDIFF	EQUALS	QERROR	# ATTITUDE ERRORS:
-------	--------	--------	--------------------

RDIFF	EQUALS	RERROR	# SCALED AT PI RADIANS.
-------	--------	--------	-------------------------

TORQUE VECTOR RECONSTRUCTION VARIABLES:

(18D)

JETRATE	EQUALS	DAPTREG1	
JETRATEQ	EQUALS	JETRATE +1	# THE LAST CONTROL SAMPLE PERIOD OF 100 MS.

JETRATER	EQUALS	JETRATE +2	# SCALED AT PI/4 RADIANS/SECOND
----------	--------	------------	---------------------------------

DOWNTORK	ERASE	+5	# ACCUMULATED JET TORQUE COMMANDED ABOUT
----------	-------	----	--

POSTORKP	EQUALS	DOWNTORK	# +,-P, +,-U, +,-V RESPECTIVELY.
----------	--------	----------	----------------------------------

NETTOTKP	EQUALS	DOWNTORK +1	# EMPLOYED EXCLUSIVELY FOR DOWNLIST.
----------	--------	-------------	--------------------------------------

POSTORKU	EQUALS	DOWNTORK +2	# NOT INITIALIZED: PERMITTED TO OVERFLOW.
----------	--------	-------------	---

```
1  NEGORKU      EQUALS  DOWNTORK +3  # SCALED AT 32 JET-SEC, OR ABOUT 2.0 JET-
2  POSTORKV     EQUALS  DOWNTORK +4  # MSEC. PER BIT.
3  NEGORKV      EQUALS  DOWNTORK +5
```

```
5  NO.PJETS     ERASE    +2
6  NO.UJETS     =        NO.PJETS +1
7  NO.VJETS     =        NO.UJETS +1
8  TJP          ERASE    +2
9  TJU          =        TJP +1
10 TJV          =        TJP +2
```

```
12 L,PVT-CG     ERASE
13 1JACC         ERASE    +4          # ACCELERATIONS DUE TO 1 JET TORQUING
14 1JACCQ        EQUALS  1JACC +1     # SCALED AT PI/4 RADIANS/SECOND
15 1JACCR        EQUALS  1JACC +2
16 1JACCU        EQUALS  1JACC +3     # FOR U,V-AXES THE SCALE FACTOR IS DOFF:
17 1JACCV        EQUALS  1JACC +4     # SCALED AT PI/2 RADIANS/SECOND (FOR ASC)
```

```
19 # ASCENT VARIABLES (10D)
```

```
21 SKIPU        ERASE    +1
22 SKIPV         =        SKIPU +1
```

```
24 # THE FOLLOWING LM DAP ERASABLES ARE ZEROED IN THE STARTDAP SECTION OF THE DAPIDLER PROGRAM AND THE COASTASC
25 # SECTION OF THE AOSTASK. THE ORDER MUST BE PRESERVED FOR THE INDEXING METHODS WHICH ARE EMPLOYED IN THOSE
26 # SECTIONS AND ELSEWHERE.
```

```
28 AOSQ          ERASE    +5          # OFFSET ACC. ESTIMATES, UPDATED IN D.P.,
29 AOSR          EQUALS  AOSQ +2     # AND SCALED AT PI/2.
30 AOSU          EQUALS  AOSQ +4     # UV-AXES OFFSET ACC. FROMED BY VECTOR
31 AOSV          EQUALS  AOSQ +5     # ADDITION OF Q,R. AT PI/2 RAD/SEC(2).
```

```
33 AOSQTERM      ERASE    +1          # (.1-.05K)AOS
34 AOSRTERM      EQUALS  AOSQTERM +1 # SCALED AT PI/4 RADIANS/SECOND.
```

```
36 # FOR TJET LAW SUBROUTINES: (TEMPS ONLY)
```

```
38 #NUMBERT      EQUALS  DAPTEMP5     # DEFINED IN QRAXIS.
```

```
39 EDOTSQ        EQUALS  DAPTEMP1
40 ROTSENSE      EQUALS  DAPTEMP2
41 FIREFCT       EQUALS  DAPTEMP3     # LOOKED AT BY PAXIS.
```

```
42 TTOAXIS       EQUALS  DAPTEMP4
43 ADRSDIF2      EQUALS  DAPTEMP6
44 HOLDQ         EQUALS  DAPTREG1
```

```
45 ADRSDIF1      EQUALS  DAPTREG2
46 HH            EQUALS  DAPTREG3     # DOUBLE PRECISION.
47 # HH +1       EQUALS  DAPTREG4
```

```
48 E             EQUALS  DAPTREG6     # TIME SHARE WITH VERROR
49 EDOT          EQUALS  OMEGAV
```


INPUT TO TJET LAW (PERMANENT ERASABLES). (48D)

TJETU	=	TJU	# EQUATE NAMES. INDEXED BY -1, 0, +1.
BLOCKTOP	ERASE	+47D	
1/ANET1	=	BLOCKTOP +16D	# THESE 8 PARAMETERS ARE SET UP BY 1/ACCS
1/ANET2	=	1/ANET1 +1	# FOR MINIMUM JETS ABOUT THE U-AXIS WHEN
1/ACOAST	=	1/ANET1 +4	# EDOT IS POSITIVE. TJETLAW INDEXES BY
ACCFCTZ1	=	1/ANET1 +6	# ADRSDIFF FROM THESE REGISTERS TO PICK UP
ACCFCTZ5	=	1/ANET1 +7	# PARAMETERS FOR THE PROPER AXIS, NUMBER
FIREDB	=	1/ANET1 +10D	# OF JETS AND SIGN OF EDOT. THERE ARE 48
COASTDB	=	1/ANET1 +12D	# REGISTERS IN ALL IN THIS BLOCK.
AXISDIST	=	1/ANET1 +14D	# FOUR NOT REFERENCED (P-AXIS) ARE FILLED
			# IN BY THE FOLLOWING:
ACCSWU	=	BLOCKTOP	# SET BY 1/ACCS TO SHOW WHETHER MAXIMUM
ACCSWV	=	ACCSWU +1	# JETS ARE REQUIRED BECAUSE OF AOS.
FLAT	=	BLOCKTOP +6	# WIDTH OF MINIMUM IMPULSE ZONE.
ZONE3LIM	=	BLOCKTOP +7	# HEIGHT OF MINIMUM IMPULSE ZONE (AT 4 SEC.)

COEFFQ	ERASE	+1	# COEFFQ AND COEFFR ARE USED IN ROT-TOUV
COEFFR	EQUALS	COEFFQ +1	# TO REXOLVE Q,R COMPONENTS INTO U,V COMP.

VARIABLES FOR GTS-QRAXIS CONTROL EXCHANGE. (4)

ALLOWGTS	EQUALS	NEGUQ +1	# INSERT INTO UNUSED LOCATION
COTROLER	ERASE		# INDICATES WHICH CONTROL SYSTEM TO USE.
QGIMTIMR	ERASE	+2	# Q-GIMBAL DRIVE ITMER, DECISECONDS.
INGTS	EQUALS	QGIMTIMR +1	# INDICATOR OF CURRENT GTS CONTROL.
RGIMTIMR	EQUALS	QGIMTIMR +2	# R-GIMBAL DRIVE TIMER, DECISECONDS.

PLEASE RETAIN THE ORDER OF CDUXD THRU CDUZD FOR DOWNLINK PURPOSES.

KALCMANU:DAP INTERFACE (9D)

CDUXD	ERASE	+2	# CDU DESIRED REGISTERS:
CDUYD	EQUALS	CDUXD +1	# SCALED AT PI RADIANS (180 DEGREES)
CDUZD	EQUALS	CDUXD +2	# (STORE IN 2'S COMPLEMENT)
DELCDUX	ERASE	+2	# NEGATIVE OF DESIRED 100MS CDU INCREMENT:
DELCDUY	EQUALS	DELCDUX +1	# SCALED AT PI RADIANS (180 DEGREES)
DELCDUZ	EQUALS	DELCDUX +2	# (STORE IN 2'S COMPLEMENT)

RETAIN THE ORDER OF OMEGAPD TO OMEGARD FOR DOWNLINK PURPOSES.

OMEGAPD	ERASE	+2	# ATTITUDE MANEUVER DESIRED RATES:
OMEGAQD	EQUALS	OMEGAPD +1	# (NOT EXPLICITLY REFERENCED IN GTS CNTRL)
OMEGARD	EQUALS	OMEGAPD +2	# SCALED AT PI/4 RADIANS/SECOND

KALCMANU STORAGE. (24D)

MIS	ERASE	+23D	# I(18D)
-----	-------	------	----------

COF	EQUALS	MIS +18D	# I(6)
# KALCMANU STORAGE. (33D)			
BCDU	ERASE	+30D	# B(3)
KSPNDX	EQUALS	BCDU +3	# B(1)
KDPNDX	EQUALS	KSPNDX +1	# B(1)
TMIS	EQUALS	KDPNDX +1	# I(18) MUST BE IN SAME BANK AS RCS DAP
COFSKEW	EQUALS	TMIS +18D	# I(6) MUST BE IN THE SAME BANK AS RCS DAP
CAM	EQUALS	COFSKEW +6	# I(2) MUST BE IN THE SAME BANK AS RCS DAP
AM	ERASE	+1	# I(2) THIS WAS ONCE IN E5 OVERLAYING OGC
# FIRST-ORDER OVERLAYS IN KALCMANU (25D)			
KV1	EQUALS	TMIS	# I(6)
MFISYM	EQUALS	TMIS	# I
TMFI	EQUALS	TMIS	# I
NCDU	EQUALS	TMIS	# B
NEXTIME	EQUALS	TMIS +3	# B
TTEMP	EQUALS	TMIS +4	# B
KV2	EQUALS	TMIS +6	# I(6)
BIASTEMP	EQUALS	TMIS +6	# B
KV3	EQUALS	TMIS +12D	# I(6)
OGF	EQUALS	TMIS +12D	# I
BRATE	EQUALS	COFSKEW	# B
IG	EQUALS	COFSKEW	# I
TM	EQUALS	CAM	# B
# SECOND-ORDER OVERLAYS IN KALCMANU (24D)			
K1	=	KV1	
K2	=	KV2	
K3	=	KV3	
P21	EQUALS	KV1	# I(2)
D21	EQUALS	KV1 +2	# I(2)
G21	EQUALS	KV1 +4	# I(2)
C2SQP	EQUALS	KV2	# I(2)
C2SQM	EQUALS	KV2 +2	# I(2)
C2PP	EQUALS	KV2 +4	# I(2)
C2MP	EQUALS	KV3	# I(2)
C1PP	EQUALS	KV3 +2	# I(2)
C1MP	EQUALS	KV3 +4	# I(2)

ERASABLE_ASSIGNMENTS

VECQTEMP = COFSKEW

DCDU = CDUXD

DELDCDU = DELCDUX

DELDCDU1 = DELCDUY

DELDCDU2 = DELCDUZ

* * * * *

STORAGE FOR FINDCDUW

OVERLAYING KALCMANU STORAGE: (26D)

ECDUW EQUALS MIS

ECDUWUSR EQUALS ECDUW # B(1)TMP

QCDUWUSR EQUALS ECDUWUSR +1 # I(1)TMP

NDXCDUW EQUALS QCDUWUSR +1 # B(1)TMP

FLAGOODW EQUALS NDXCDUW +1 # B(1)TMP

FLPAUTNO EQUALS FLAGOODW +1 # B(1)TMP

UNFC/2 EQUALS FLPAUTNO +1 # I(6)IN

UNWC/2 EQUALS UNFC/2 +6 # I(6)IN

UNFV/2 EQUALS UNWC/2 +6 # I(6) S-S

UNFVX/2 = UNFV/2

UNFVY/2 = UNFV/2 +2

UNFVZ/2 = UNFV/2 +4

-DELGMB EQUALS UNFV/2 +6 # B(3)TMP

DEFINED IN THE WORK AREA: (18D)

UNX/2 = 0

UNY/2 = 6

UNZ/2 = 14

END OF FINDCDUW ERASABLES

* * * * *

THE FOLLOWING ARE THE DAP REPLACEMENTS FOR THE ITEMPS AND RUPTREGS, NEEDED BECAUSE DAP IS NOW A TOB,JASK,JAB,TOSK

... ANYWAY, THE DAP CAN NOW BE INTERRUPTED. (18D)

DAPTEMP1 ERASE +17D

DAPTEMP2 EQUALS DAPTEMP1 +1

DAPTEMP3 EQUALS DAPTEMP1 +2

DAPTEMP4 EQUALS DAPTEMP1 +3

DAPTEMP5 EQUALS DAPTEMP1 +4

DAPTEMP6 EQUALS DAPTEMP1 +5

DAPTREG1 EQUALS DAPTEMP1 +6

DAPTREG2 EQUALS DAPTEMP1 +7

DAPTREG3 EQUALS DAPTEMP1 +8D

ERASABLE_ASSIGNMENTS

1				1
2	DAPTREG4	EQUALS	DAPTEMP1 +9D	2
3	DAPTREG5	EQUALS	DAPTEMP1 +10D	3
4	DAPTREG6	EQUALS	DAPTEMP1 +11D	4
5				5
6	DAPARUPT	EQUALS	DAPTEMP1 +12D	6
7	DAPLRUPT	EQUALS	DAPARUPT +1	7
8	DAPBQRPT	EQUALS	DAPARUPT +2	8
9	DAPZRUPT	EQUALS	DAPARUPT +4	9
10	# (DAPZRUPT IS ALSO A JASK-IN-PROGRESS FLAG)			10
11				11
12	# NEEDLER (ATTITUDE ERROR EIGHT BALL DISPLAY) STORAGE. (6D)			12
13				13
14	T5TEMP	EQUALS	ITEMP1	14
15	DINDX	EQUALS	ITEMP3	15
16	AK	ERASE	+2	16
17	AK1	EQUALS	AK +1	17
18	AK2	EQUALS	AK +2	18
19	# NEEDLER ATTITUDE INPUTS, SCALED AT 180			19
20	EDRIVEX	ERASE	+2	20
21	EDRIVEY	EQUALS	EDRIVEX +1	21
22	EDRIVEZ	EQUALS	EDRIVEX +2	22
23	# SO THAT 384 BITS REPRESENT 42 3/16 DEGREES.			23
24	# DOCKED JET INHIBITION COUNTERS (3D)			24
25				25
26	PJETCTR	ERASE	+2	26
27	UJETCTR	EQUALS	PJETCTR +1	27
28	VJETCTR	EQUALS	PJETCTR +2	28
29				29
30	END-E6	EQUALS	VJETCTR	30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1412THE

1

EBANK-7 ASSIGNMENTS

SETLOC 3400

P35 CONSTANTS. -- PAD LOADED -- (4D)

ATIGINC	ERASE	+1	# B(2)PL	*MUST BE AT 1400 FOR SYSTEMSTEST
PTIGINC	ERASE	+1	# B(2)PL	

AOTMARK STORAGE. -- PAD LOADED -- (12D)

AOTAZ	ERASE	+5	# B(6)PL
AOTEL	ERASE	+5	# B(6)PL

LANDING RADAR -- PAD LOADED -- (2D)

LRHMAX	ERASE	# B(1)
LRWH	ERASE	# B(1)

THROTTLE STORAGE. -- PAD LOADED -- (1D)

ZOOMTIME ERASE # B(1)PL TIME OF DPS THROTTLE-UP COMMAND

P63 AND P64 CONSTANTS. -- PAD LOADED -- (4D)

TENDBRAK	ERASE	# B(1) LANDING PHASE SWITCHING CRITERION.
TENDAPPR	ERASE	# B(1) LANDING PHASE SWITCHING CRITERION.
DELTTFAP	ERASE	# B(1) INCREMENT ADDED TO TTF/8 WHEN SWITCHING FROM P63 TO P64.
LEADTIME	ERASE	# B(1) TIME INCREMENT SPECIFYING HOW MUCH GUIDANCE IS PROJECTED FORWARD

LANDING RADAR. -- PAD LOADED -- (2D)

RPCRTIME	ERASE	# B(1) REPOSITIONING CRITERION (TIME)
RPCRTQSW	ERASE	# B(1) REPOSITIONING CRITERION (ANGLE)

ASTEER. -- PAD LOADED -- (2D)

TNEWA ERASE +1 # I(2)PL LAMBERT CYCLE PERIOD.

P22 STORAGE -- OVERLAYS LANDING PADLOADS -- (5D)

```
1 REPOSCNT      EQUALS  TENDBRAK      # B(1)TMP COUNTS NUMBER OF PASSES THROUGH
2                                     # REPOSITION ROUTINE.
3
4 REPOSTM       EQUALS  REPOSCNT +1    # I(2)TMP PRESENT TIME PLUS INCREMENTS OF
5                                     # TEN SECONDS.
6 DELTATM       EQUALS  REPOSTM +2     # I(2)TMP TIME INTERVAL FOR RUNNING
7                                     # DESIGNATE TASK.
8
9 # *** RETAIN THE ORDER OF DELVSLV, TIG, RTARG, DELLT4 FOR UPDATE. ***
10
11 # P32-35 P72-75 STORAGE.              (6D)
12
13 DELVLVC       ERASE   +5              # I(6) DELTA VELOCITY -- LOCAL VERTICAL COO
14 DELVSLV       =      DELVLVC         # (TEMP STORAGE OF SAME VECTOR)  -RDINATE
15
16 # P30-P40 INTERFACE UNSHARED.          (2D)
17
18 TIG           ERASE   +1              # B(2)
19
20 # INITVEL STORAGE.  ALSO USED BY P34,35,74,75,10,11 OTHERS      (8D)
21
22 RTARG         ERASE   +5              # I(6) TARGET VECTOR
23 DELLT4        ERASE   +1              # I(2) TIME DIFFERENCE
24
25 # P30-P40 INTERFACE UNSHARED.          (3D)
26
27 TTOGO         ERASE   +1              # B(2)
28 TFI           EQUALS  TTOGO
29 WHICH         ERASE
30
31 # *** R21 ***                          (1D)
32
33 LOSCOUNT     ERASE                  # B(1)
34
35 # L SR22.3 (RENDEZVOUS NAVIGATION) STORAGE.      (4D)
36
37 # RETAIN THE ORDER OF AIG TO TRKMKCNT FOR DOWNLINK PURPOSES.
38
39 AIG           ERASE                  # B(1)OUT GIMBAL ANGLES
40 AMG           ERASE                  # B(1)OUT (MUST BE
41 AOG           ERASE                  # B(1)OUT CONSECUTIVE)
42
43 TRKMKCNT      ERASE                  # B(1)TMP TEMPORARY MARK STORAGE.
44 MARKCTR      =      TRKMKCNT
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

ERASABLE_ASSIGNMENTS

P32-P35, P72-P75 STORAGE. -- PERMANENT -- (6)

NORMEX	ERASE		# B(1)PRM SAVE FOR Q
QSAVED	ERASE		# B(1)PRM SAVE FOR Q
RTRN	ERASE		# B(1)PRM SAVE FOR Q
NN	ERASE	+1	# B(2)
SUBEXIT	ERASE		# B(1)PRM SAVE Q
E7OVERLA	EQUALS		# START OF E7 OVERLAYS
WHOCARES	EQUALS	E7OVERLA	# DUMMY FOR EBANK INSENSITIVE 2CADRS.

LUNAR LANDING OVERLAYS (6D)

/AFC/	EQUALS	NORMEX	# B(2)TMP THROTTLE
FCODD	EQUALS	/AFC/ +2	# B(2)TMP THROTTLE
FP	EQUALS	FCODD +2	# B(2)TMP THROTTLE

***** OVERLAY NUMBER 0 IN EBANK 7 *****

RENDEZVOUS GUIDANCE STORAGE --P32.....P35-- (89D)

TSTRT	EQUALS	DELDV	# MIDCOURSE START TIME
TDEC2	EQUALS	DELVCSI	# TEMP STORAGE FOR INTEGRATION TIME INPUT
KT	EQUALS	DELVTPI	# TEMP STORAGE FOR MIDCOURSE DELTA TIME
VACT1	ERASE	+5D	# VELOCITY VECTOR OF ACTIVE AT CSI TIME
RPASS1	ERASE	+5D	# POSITION VECTOR OF PASSIVE AT CSI TIME
VPASS1	ERASE	+5D	# VELOCITY VECTOR OF PASSIVE AT CSI TIME
VACT2	ERASE	+5D	# VELOCITY VECTOR OF ACTIVE AT CDH TIME
RPASS2	ERASE	+5D	# POSITION VECTOR OF PASSIVE AT CDH TIME
VPASS2	ERASE	+5D	# VELOCITY VECTOR OF PASSIVE AT CDH TIME
RACT3	ERASE	+5D	# POSITION VECTOR OF ACTIVE AT TPI TIME
VACT3	ERASE	+5D	# VELOCITY VECTOR OF ACTIVE AT TPI TIME
RPASS3	ERASE	+5D	# POSITION VECTOR OF PASSIVE AT TPI TIME
VPASS3	ERASE	+5D	# VELOCITY VECTOR OF PASSIVE AT TPI TIME
VACT4	ERASE	+5D	# VELOCITY VECTOR OF ACTIVE AT INTERCEPT
UNVEC	EQUALS	VACT3	# CDHMVR UNIT VECTOR TEMP STORAGE.
DELVCSI	ERASE	+1D	# THRUST VALUE AT CSI
DELVTPI	ERASE	+1D	# THRUST VALUE AT TPI OR MID
DELMID	EQUALS	DELVTPI	
DIFFALT	ERASE	+1D	# ALT DIFFERENT AT CDH
POSTCSI	ERASE	+1	# PERIGEE ALTITUDE AFTER CSI MANEUVER
POSTCDH	ERASE	+1	# PERIGEE ALTITUDE AFTER CDH MANEUVER
POSTTPI	ERASE	+1	# PERIGEE ALTITUDE AFTER TPI MANEUVER
LOOPCT	EQUALS	POSTTPI	# CSI NEWTON ITERATION COUNTER
HAFPA1	EQUALS	POSTCDH	# HALF PERIOD
GAMPREV	ERASE	+1	# PREVIOUS GAMMA
DVPREV	EQUALS	DELVTPI	# PREVIOUS DELVCSI
DELDV	ERASE	+1D	
CSIALRM	ERASE	+1	# FIRST SOLUTION ALARM
VERBNOUN	ERASE		
TITER	EQUALS	CSIALRM	# ITERATION COUNTER
RDOTV	ERASE	+1	
VAPREC	EQUALS	VPASS1	# I(6) S-S PREC VEC FOR NOM TPI TIME (ACTIVE)
RAPREC	EQUALS	RPASS1	# I(6) S-S PREC VEC FOR NOM TPI TIME (ACTIVE)
VPPREC	EQUALS	VPASS2	# I(6) S-S PREC VEC FOR NOM TPI TIME (PASSIVE)
RPPREC	EQUALS	RPASS2	# I(6) S-S PREC VEC FOR NOM TPI TIME (PASSIVE)
DELEL	EQUALS	DELVTPI	# I(2) S-S
DELTEE	EQUALS	DELDV	# I(2) S-S
SECMAX	EQUALS	DELVCSI	# I(2) S-S MAX STOP SIZE FOR ROUTINE
DELTEEO	EQUALS	POSTTPI	# I(2) S-S BACK VALUES OF DELTA TIME
CENTANG	ERASE	+1	# I(2) CENTRAL ANGLE COVERED (TPI-TPF)

```
1  # SOME P47 STORAGE (6D)
2
3
4  DELVIMU          ERASE   +5          # I(6)DSP NOUN 83 FOR P47 DELTA V (IMU)
5
6  # P30-P40 COMMON STORAGE. (3D)
7
8  TPASS4           ERASE   +1          # INTERCEPT TIME
9  QTEMP            ERASE
10
11 # P32,33,34 STORAGE. (6D)
12
13 TCSI              ERASE   +1          # B(2)TMP CSI TIME IN CENTISECONDS
14 TTPI              ERASE   +1          # B(2)TMP TPI TIME IN CENTISECONDS
15 TTPIO             ERASE   +1          # B(2)TMP TTPI STORAGE FOR RECYCLE
16
17 # P30,P40 INTERFACE. (21D)
18
19 RTIG              ERASE   +19D        # I(6)TMP
20 VTIG              EQUALS  RTIG +6      # I(6)TMP
21 DELVSIN           EQUALS  VTIG +6      # I(6)TMP
22 DELVSAB           EQUALS  DELVSIN +6   # I(2)TMP
23 VGDISP            =       DELVSAB
24
25 QTEMP1            ERASE
26 RGEXIT            EQUALS  QTEMP1      # I(1)TMP HOLDS RETURN.
27 SAVQR52           EQUALS  QTEMP1      # SAVE Q
28
29 # INITVEL STORAGE. (IN OVERLAY 0 AND OVERLAY 1. (2D)
30 # (CALLS LAMBERT, CONIC SUBROUTINES)
31
32 VTPRIME           EQUALS  VACT4        # TOTAL VELOCITY AT DESIRED RADIUS
33 ITCTR             EQUALS  RDOTV        # ITERATION COUNTER
34 COZY4             ERASE   +1          # COS OF ANGLE WHEN ROTATION STARTS
35 X1INPUT           EQUALS  DELDV        # X1 TEMP STORAGE
36 INTIME            EQUALS  GAMPREV      # TIME OF RINIT
37
38 # PERIAPO STORAGE. (2D)
39
40 XXXALT            ERASE   +1          # RADIUS TO LAUNCH PAD OR LANDING SITE
41
42 END-IN/M          EQUALS  XXXALT +2    # NEXT AVAIL ERASABLE AFTER INITVEL/MIDGIM
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```


ERASABLE_ASSIGNMENTS

S40.1 STORAGE. (12D)

UT ERASE +11D # I(6) THRUST DIRECTION
VGTIG EQUALS UT +6 # I(6)OUT
VGPREV = VGTIG

ASTEER STORAGE. (22D)

VG ERASE +21D # I(6)
RMAG EQUALS VG +6 # I(2)
MUASTEER EQUALS RMAG +2 # I(2)
MU/A EQUALS MUASTEER +2 # I(2)
RTMAG EQUALS MU/A +2 # I(2)
RIC EQUALS RTMAG +2 # I(6)
SS EQUALS RIC +6 # I(2)

IC = DELVSIN
TIGSAVE = P21TIME
TIGSAVEP = SCAXIS
MUSCALE = SCAXIS +2

P40 STORAGE. (6D)

F, MDOT, AND TDECAY MUST BE CONTIGUOUS FOR WLOAD

F ERASE +5 # I(2)TMP S40.1 GENERATES THIS FOR S40.3
MDOT EQUALS F +2 # I(2)TMP MASS CHNG RATE, KG/CS AT 2**3.
TDECAY EQUALS MDOT +2 # I(2)IN DELTA-T TAILOFF, (2**28)CS.
VEX ERASE +1 # I(2) EXHAUST VELOCITY FOR TGO COMPUTAT'N

MIDTOAV1(2) STORAGE. (CALLED BY P40,P41,P42) (1D)

IRETURN1 ERASE # B(1) RETURN FROM MIDTOAV1 AND 2.

```
# INITVEL (CALLED BY P34,35,38,39,10,11,S40.9,S40.1) (6D)
```

```
RTARG1          EQUALS  VACT1          # I(6)S TEMP STORAGE OF RTARG
```

P35-P40 INTERFACE. (6D)

VPASS4	EQUALS	VPASS1	# I(6)TMP VELOCITY OF PASSIVE AT INTERCEPT
1	1	1	1

```
# INITVEL OVERLAYS RENDESVOUS GUIDANCE (LISTED IN OVERLAY 0)
```

```
# SOME P38-30,P78-79 STORAGE (2D)
TINT          EQUALS    TPASS4      # I(2) TIME OF INTERCEPT
```

```
# LAT-LONG TEMPORARIES.  CAN OVERLAY WITH S40.1          (3D)
```

```
ERADM          EQUALS  UT          # I(2)
INCORPEX      EQUALS  ERADM +2     # I(1)
```

LRS24.1 STORAGE. (CAN SHARE WITH P30'S) (40D)

RLMSRCH	EQUALS	INCORPEX +1	# I(6)TMP	LM POSITION VECTOR
VXRCM	EQUALS	RLMSRCH +6	# I(6)	CM V X R VECTOR
LOSDESRD	EQUALS	VXRCM +6	# I(6)	DESIRED LOS VECTOR
UXVECT	EQUALS	LOSDESRD +6	# I(6)	X-AXIS SRCH PATTERN COORDS
UYVECT	EQUALS	UXVECT +6	# I(6)	Y-AXIS SRCH PATTERN COORDS
DATAGOOD	EQUALS	UYVECT +6	# B(1)DSP	FOR R1 -- ALL 1-S WHEN LOCKON
OMEGDISP	EQUALS	DATAGOOD +1	# B(2)	ANGLE OMEGA DISPLAYED IN R2
OMEGAD	=	OMEGDISP	#	PINBALL DEFINITION
NSRCHPNT	EQUALS	OMEGDISP +2	# B(1)TMP	SEARCH PATTERN POINT COUNTER.
SAVLEMV	EQUALS	NSRCHPNT +1	# I(6)S-S	SAVES LOSVEL

***** OVERLAY NUMBER 2 IN EBANK 7 *****

INCORP STORAGE IN E7. (47D)

TX789 EQUALS E7OVERLA # I(6)

GAMMA EQUALS TX789 +6 # I(3)

OMEGA EQUALS GAMMA +3 # I(18)

BVECTOR EQUALS OMEGA +18D # I(18)

DELTAQ EQUALS BVECTOR +18D # I(2)

AOTMARK STORAGE (3D)

MARKCNTR EQUALS DELTAQ +2 # I(1)

XYMARK EQUALS MARKCNTR +1 # B(1)

MKDEX EQUALS XYMARK +1 # B(1)TMP INDEX FOR AOTMARK

PLANET STORAGE (8D)

PLANVEC EQUALS MKDEX +1 # (6) REFER VECTOR OF PLANET

TSIGHT EQUALS PLANVEC +6 # (2) TIME OF MARK OR EST TIME OF MARK

LRS22.3 STORAGE. (CAN SHARE WITH P30'S AND OVERLAY LRS24.1) (30D)

LGRET EQUALS RLMSRCH # I(1)TMP

RDRET EQUALS LGRET # B(1) TEMP RETURN.

IGRET EQUALS LGRET # B(1) TEMP RETURN.

MX EQUALS RDRET +1 # I(6)

MY EQUALS MX +6 # I(6)

MZ EQUALS MY +6 # I(6)

E0 EQUALS MX # I(2)

E1 EQUALS MX +2 # I(2)

E2 EQUALS MX +4 # I(2)

E3 EQUALS E2 +2 # I(2)

SCALSHFT EQUALS MZ +6 # B(1) SCALE SHIFT FOR EARTH/MOON

RXZ EQUALS SCALSHFT +1 # I(2)

ULC EQUALS RXZ +2 # I(6)

SINTHETA EQUALS ULC +6 # I(2)

***** IN OVERLAY ONE *****

N49FLAG EQUALS RDOTMSAV # B(1)S FLAG INDICATING V0649 RESPONSE

LRS22.1 STORAGE. (MUST NOT SHARE WITH P30'S) (13D)

(OUTPUTS ARE TO LRS22.3)

```
1 RRTRUN      EQUALS  SINTHETA +2    # B(2)OUT RR TRUNNION ANGLE
2 RRSHAFT     EQUALS  RRTRUN +2      # B(2)OUT RR SHAFT ANGLE
3 LRS22.1X    EQUALS  RRSHAFT +2     # B(1)TMP
4 RRBORSIT    EQUALS  LRS22.1X +1    # I(6)TMP RADAR BORESIGHT VECTOR.
5 RDOTMSAV    EQUALS  RRBORSIT +6    # B(2)S RR RANGE-RATE (FPS)
```

```
6
7
8 # LRS22.1 (SAME AS PREVIOUS SECTION) ALSO DOWNLINK FOR RR (R29)          (10D) CANNOT SHARE WITH L.A.D.
```

```
9
10 RDOTM       EQUALS  RDOTMSAV +2    # B(2)OUT RANGE-RATE READING
11 TANGNB      EQUALS  RDOTM +2       # B(2)TMP RR GIMBAL ANGLES
12 # RETAIN THE ORDER OF MKTIME TO RM FOR DOWNLINK PURPOSES
13 MKTIME      EQUALS  TANGNB +2      # B(2)OUT TIME OF RR READING
14 RM          EQUALS  MKTIME +2      # I(2)OUT RANGE READING
15 RANGRDOT    EQUALS  RM +2          # B(2) DOWNLINKED RAW RANGE AND RRATE
```

```
16
17 # R61LEM -- PREFERRED TRACKING ATTITUDE ROUTINE **IN OVERLAY ONE**
18 # (CALLED BY P20, R22LEM, LSR22.3)          (1D)
```

```
19
20 R65CNTR      EQUALS  RRBORSIT +5    # B(1)SS COUNT NUMBER OF TIMES PREFERRED
21                                     # TRACKING ROUTINE IS TO CYCLE
22 WHCHREAD     EQUALS  R65CNTR       # TELLS WHICH RR DATA TRIGGERED N49 DISPLAY
```

```
23
24 # P21 STORAGE          (2D)
```

```
25
26 P21TIME      EQUALS  RANGRDOT +2    # I(2)TMP
```

```
27
28 # KALCMANU, VECPOINT STORAGE. CALLED BY R63, R61, R65.          (12D)
```

```
29
30 SCAXIS       EQUALS  P21TIME +2     # I(6)
31 POINTVSM     EQUALS  SCAXIS +6      # I(6)
```

***** OVERLAY NUMBER 3 IN EBANK 7 *****

SERVICER STORAGE (6D)

ABVEL	EQUALS	E7OVERLA	# B(2) DISPLAY
HDOTDISP	EQUALS	ABVEL +2	# B(2) DISPLAY
TTFDISP	EQUALS	HDOTDISP +2	# B(2) DISPLAY

BURN PROG STORAGE. (2D)

SAVET-30 EQUALS TTFDISP +2 # B(2)TMP TIG-30 RESTART

SERVICER STORAGE. (69D)

VGBODY	EQUALS	SAVET-30 +2	# B(6)OUT SET BY S41.1 VG LEM, SC.COORDS
DELVCTL	=	VGBODY	
DVTOTAL	EQUALS	VGBODY +6	# B(2) DISPLAY NOUN
GOBLTIME	EQUALS	DVTOTAL +2	# B(2) NOMINAL TIG FOR CALC. OF GOBLATE.
ABDVCONV	EQUALS	GOBLTIME +2	# I(2)
DVCNTR	EQUALS	ABDVCONV +2	# B(1)
TGO	EQUALS	DVCNTR +1	# B(2)
R	EQUALS	TGO +2	# I(6)
UNITGOBL	EQUALS	R	# I(6)
V	EQUALS	R +6	
DELVREF	EQUALS	V	# I(6)
HCALC	EQUALS	DELVREF +6	# B(2) LR
UNIT/R/	EQUALS	HCALC +2	# I(6)

(THE FOLLOWING SERVICER ERASABLES CAN BE SHARED WITH SECOND DPS GUIDANCE STORAGE)

RN1	EQUALS	UNIT/R/ +6	# B(6)	
VN1	EQUALS	RN1 +6	# I(6)	(IN ORDER)
PIPTIME1	EQUALS	VN1 +6	# B(2)	(FOR)
GDT1/2	EQUALS	PIPTIME1 +2	# I(6)	(COPY)
MASS1	EQUALS	GDT1/2 +6	# I(2)	(CYCLE)
R1S	EQUALS	MASS1 +2	# I(6)	
V1S	EQUALS	R1S +6	# I(6)	

ALIGNMENT/S40.2,3 COMMON STORAGE. (18D)

XSMD	EQUALS	V1S +6	# I(6)
YSMD	EQUALS	XSMD +6	# I(6)
ZSMD	EQUALS	YSMD +6	# I(6)

XSCREF	=	XSMD
YSCREF	=	YSMD

ERASABLE_ASSIGNMENTS

1						1
2	ZSCREF	=	ZSMD			2
3						3
4	END-ALIG	EQUALS	ZSMD +6	# NEXT AVAIL ERASABLE AFTER ALIGN/S40.2,3		4
5						5
6	# ***** P22 *****		(24D)			6
7						7
8	RSUBL	EQUALS	END-ALIG	# I(6)S-S	LM POSITION VECTOR	8
9	UCSM	EQUALS	RSUBL +6	# I(6)S-S	VECTOR U	9
10	NEWVEL	EQUALS	UCSM +6	# I(6)S-S	TERMINAL VELOCITY VECTOR	10
11	NEWPOS	EQUALS	NEWVEL +6	# I(6)S-S	TERMINAL POSITION VECTOR	11
12	LNCHTM	EQUALS	NEWPOS +6	# I(2)S-S	EST. LAUNCH TIME FOR LEM	12
13	TRANSTM	EQUALS	LNCHTM +2	# I(2)S-S	TRANSFER TIME	13
14	NCSMVEL	EQUALS	TRANSTM +2	# I(6)S-S	NEW CSM VELOCITY	14
15						15
16	# ***** P21 *****		(18D)			16
17						17
18	P21ORIG	=	DISPDEX			18
19	P21BASER	EQUALS	RLMSRCH	# I(6)TMP		19
20	P21BASEV	EQUALS	P21BASER +6	# I(6)TMP		20
21	P21VEL	EQUALS	P21BASEV +6	# I(2)TMP	*** NOUN 91 ***	21
22	P21GAM	EQUALS	P21VEL +2	# I(2)TMP	*** NOUN 91 ***	22
23	P21ALT	EQUALS	P21GAM +2	# I(2)TMP	*** NOUN 91 ***	23
24						24
25						25
26						26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

***** OVERLAY NUMBER 4 IN EBANK 7 *****

VARIABLES FOR SECOND DPS GUIDANCE (THE LUNAR LANDING) (18D)

THESE ERASABLES MAY BE SHARED WITH CARE

OURTEMPS	=	RN1	# OVERLAY LAST PART OF SERVICER
LANDTEMP	=	OURTEMPS	# B(6) GUIDANCE
TTF/8TMP	=	LANDTEMP +6	# B(2) GUIDANCE
ELINCR	=	TTF/8TMP +2	# B(2) GUIDANCE
AZINCR	=	ELINCR +2	# B(2) GUIDANCE
KEEP-2	=	AZINCR +2	# B(2) TP PREVENT PIPTIME1 OVERLAY
TABLTTF	=	KEEP-2 +2	# B(2) GUIDANCE
TPIPOLD	=	TABLTTF +9D	# B(2) GUIDANCE
E2DPS	EQUALS	OURPERMS	

THESE ERASABLES MUST NOT OVERLAY GOBLTIME OR SERVICER

PIFPSET	=	XSMD	# B(1) THROTTLE
RTNHOLD	=	PIFPSET +1	# B(1) THROTTLE
FWEIGHT	=	RTNHOLD +1	# B(2) THROTTLE
PIF	=	FWEIGHT +2	# B(2) THROTTLE
PSEUDO55	=	PIF +2	# B(1) THROTTLE DOWNLINK
FC	=	PSEUDO55 +1	# B(2) THROTTLE
TTHROT	=	FC +2	# B(1) THROTTLE
FCOLD	=	TTHROT +1	# B(1) THROTTLE

THESE ERASABLES SHOULD NOT BE SHARED DURING P63, P64, P65, P66, P67

OURPERMS	=	FCOLD +1	# MUSTN'T OVERLAY OURTEMPS OR SERVICER
WCHPHOLD	=	OURPERMS	# B(1) GUIDANCE
FILLER	=	WCHPHOLD +1	
FLPASS0	=	FILLER +1	# B(1) GUIDANCE
TPIP	=	FLPASS0 +1	# B(2)
VGU	=	TPIP +2	# B(6) GUIDANCE
LAND	=	VGU +6	# B(6) GUIDANCE CONTIGUOUS
TTF/8	=	LAND +6	# B(2) GUIDANCE CONTIGUOUS
ELIDUMMY	=	TTF/8 +2	# (1) DUMMY FOR ELINCR1
AZIDUMMY	=	ELIDUMMY +1	# (1) DUMMY FOR AZINCR1
ZERDUMMY	=	AZIDUMMY +1	# (1) DUMMY FOR ZERLINA
ELVDUMMY	=	ZERDUMMY +1	# (1) DUMMY FOR ELVIRA
LRADRET	=	ELVDUMMY +1	# B(1) LR
VSELECT	=	LRADRET +1	# B(1) LR
VMEAS	=	VSELECT +1	# B(2) LR
HMEAS	=	VMEAS +2	# B(2) LR
VN2	=	HMEAS +2	# B(6) LR

```
1 GNUR      =      VN2      # B(6) LR
2 GNUV      =      VN2      # B(6) LR
3 LRADRET1  =      VN2      # B(1) LR
4 DELTAH    =      VN2 +6    # B(2) DISPLAY
5 FUNNYDSP  =      DELTAH +2  # B(2) DISPLAY
6 EOURPERM  EQUALS  FUNNYDSP +2 # NEXT AVAILABLE ERASABLE AFTER OURPERMS
```

(ERASABLES WHICH OVERLAY THE ABOVE BLOCK)

```
11 VDGVERT   =      ELIDUMMY   # B(2) P65,P66
12 NIGNLOOP  =      ZERDUMMY   # B(1) IGNALG
13 NGUIDSUB  =      ELVDUMMY   # B(1) IGNALG
14 WCHVERT   =      ELVDUMMY   # B(1) P65,P66,P67
15 FUELNEED  =      FUNNYDSP    # B(1) DISPLAY
16 TREDES    =      FUNNYDSP    # B(1) DISPLAY
17 LOOKANGL  =      FUNNYDSP +1 # B(1) DISPLAY
```

ERASABLES CONVENIENTLY DEFINABLE IN THE WORK AREA

```
21 PROJ      =      18D      # I(2) GUIDANCE
22 UNLRB/2    =      20D      # I(6) GUIDANCE (DURING P64 ONLY)
23 UNLR/2     =      20D      # I(6) GUIDANCE
```

THE END OF THE LUNAR LANDING ERASABLES

R12 (FOR LUNAR LANDING) (6D)

```
29 LRLCTR    EQUALS  EOURPERM   # B(1) LR DATA TEST
30 LRRCTR    EQUALS  LRLCTR +1   # B(1)
31 LRMCTR    EQUALS  LRRCTR +1   # B(1)
32 LRSCTR    EQUALS  LRMCTR +1   # B(1)
33 STILBADH  EQUALS  LRSCTR +1   # B(1)
34 STILBADV  EQUALS  STILBADH +1 # B(1)
```

LANDING ANALOGS DISPLAY STORAGE. (40D)

```
38 LATVMETR  EQUALS  STILBADV +1 # B(1)PRM LATVEL MONITOR METER (AN ORDER)
39 FORVMETR  EQUALS  LATVMETR +1  # B(1)PRM FORVEL MONITOR METER (-ED PAIR)
40 LATVEL    EQUALS  FORVMETR +1  # B(1)PRM LATERAL VELOCITY (AN ORDER)
41 FORVEL    EQUALS  LATVEL +1    # B(1)PRM FORWARD VELOCITY (-ED PAIR)
42 TRAKLATV  EQUALS  FORVEL +1    # B(1)PRM MONITOR FLG 4 LATVEL (AN ORDER)
43 TRAKFWDV  EQUALS  TRAKLATV +1  # B(1)PRM MONIT. FLAG FOR FORVEL (ED PAIR)
44 VHY       EQUALS  TRAKFWDV +1  # B(1)PRM VHY=VMP.UHYP (AN ORDER)
```


ERASABLE_ASSIGNMENTS

1						1
2	VHZ	EQUALS	VHY +1	# B(1)PRM VHZ=VMP.UHZP (-ED PAIR)		2
3	VVECT	EQUALS	VHZ +1	# B(3)PRM UPDATED S.P. VELOCITY VECTOR		3
4	ALTRATE	EQUALS	VVECT +3	# B(1)PRM ALTITUDE RATE IN BIT UNITS		4
5	ALTSAVE	EQUALS	ALTRATE +1	# B(2)PRM ALTITUDE IN BIT UNITS		5
6	LADQSAVE	EQUALS	ALTSAVE +2	# B(1)PRM SAVE Q IN LANDISP		6
7	DT	EQUALS	LADQSAVE +1	# B(1)PRM TIME 1 MINUS (PIPTIME +1)		7
8	DALTRATE	EQUALS	DT +1	# B(1)PRM ALTITUDE RATE ERROR CORRECTION		8
9	UHYP	EQUALS	DALTRATE +1	# B(6)PRM SM UNIT VECTOR		9
10	QAXIS	=	UHYP			10
11	UHYP	EQUALS	UHYP +6	# B(6)PRM SM UNIT VECTOR		11
12	DELVS	EQUALS	UHYP +6	# B(6)PRM DELVS = WMXR		12
13	ALTBITS	EQUALS	DELVS +6	# B(2)PRM ALTITUDE IN BIT UNITS. 2.34 FT/BIT		13
14	RUNIT	EQUALS	ALTBITS +2	# B(3)PRM SM HALF-UNIT R VECTOR		14
15	LASTLADW	EQUALS	RUNIT +2	# ONLY A TAG TO SIGNIFY LAST L.A.D. WORD		15
16						16
17	# P66 ERASABLES (R.O.D.)		(1D)			17
18						18
19	RODCOUNT	EQUALS	RUNIT +3			19
20						20
21	# P66 ERASABLES (R.O.D.)		(14D)			21
22						22
23	RODSCAL1	EQUALS	RM	# B(1)		23
24	LASTTPIP	EQUALS	RODSCAL1 +1	# I(2)		24
25	THISTPIP	EQUALS	LASTTPIP +2	# B(2)		25
26	OLDPIPAX	EQUALS	THISTPIP +2	# B(1)		26
27	OLDPIPAY	EQUALS	OLDPIPAX +1	# B(1)		27
28	OLDPIPAZ	EQUALS	OLDPIPAY +1	# B(1)		28
29	DELVRD	EQUALS	OLDPIPAZ +1	# B(6)		29
30						30
31	# NOUN 63 COMPONENT		(2D)			31
32						32
33	HCALC1	EQUALS	DELVRD +6	# I(2)		33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

ERASABLE_ASSIGNMENTS

***** OVERLAY NUMBER 5 IN EBANK 7 *****

ASCENT GUIDANCE ERASABLES. (21D)

RCO	EQUALS	END-ALIG	# I(2)TMP TARGET RADIUS AND OUT-OF-PLANE
YCO	EQUALS	RCO +2	# I(2)TMP DISTANCE, SCALED AT 2(24).
1/DV1	EQUALS	YCO +2	# B(2)TMP ATMAG
1/DV2	EQUALS	1/DV1 +2	# B(2)TMP ATMAG
1/DV3	EQUALS	1/DV2 +2	# B(2)TMP ATMAG
XRANGE	EQUALS	1/DV3 +2	# B(2)TMP
ENGOFFDT	EQUALS	XRANGE +2	# B(1)TMP
VGVECT	EQUALS	ENGFFDT +1	# I(6)OUT VELOCITY-TO-BE-GAINED.
TXO	EQUALS	VGVECT +6	# I(2)TMP TIME AT WHICH X-AXIS OVERRIDE
			# IS ALLOWED.

END OF THE ASCENT GUIDANCE ERASABLES

THE FOLLOWING CARDS KEEP THE ASSEMBLER HAPPY UNTIL THE SYMBOLS ARE DELETED FROM THE PINBALL NOUN TABLES.

END-E7.0	EQUALS	IRETURN1 +1	# FIRST UNUSED LOCATION IN E7 OVERLAY 0
END-E7.1	EQUALS	N49FLAG +1	# FIRST UNUSED LOCATION IN E7 OVERLAY 1
END-E7.2	EQUALS	POINTVSM +6	# FIRST UNUSED LOCATION IN E7 OVERLAY 2
END-E7.3	EQUALS	END-ALIG	# FIRST UNUSED LOCATION IN E7 OVERLAY 3
END-E7.4	EQUALS	3777	# ** LAST LOCATION USED IN E7 OVERLAY 4 **
END-E7.5	EQUALS	TXO +2	# FIRST UNUSED LOCATION IN E7 OVERLAY 5
END-E7	EQUALS	3777	# ** LAST LOCATION USED IN E7 **

1	# INTERNAL LEAD INS			
2				
3	SETLOC	4000		
4	COUNT*	\$\$/RUPTS	# FIX-FIX LEAD INS	
5	INHINT		# GO	
6	CAF	GOBB		
7	XCH	BBANK		
8	TCF	GOPROG		
9				
10	DXCH	ARUPT	# T6RUPT	
11	EXTEND			
12	DCA	T6ADR		
13	DTCB			
14				
15	DXCH	ARUPT	# T5RUPT - AUTOPILOT	
16	EXTEND			
17	DCA	T5ADR		
18	DTCB			
19				
20	DXCH	ARUPT	# T3RUPT	
21	CAF	T3RPTBB		
22	XCH	BBANK		
23	TCF	T3RUPT		
24				
25	DXCH	ARUPT	# T4RUPT	
26	CAF	T4RPTBB		
27	XCH	BBANK		
28	TCF	T4RUPT		
29				
30	DXCH	ARUPT	# KEYRUPT1	
31	CAF	KEYRPTBB		
32	XCH	BBANK		
33	TCF	KEYRUPT1		
34				
35	DXCH	ARUPT	# KEYRUPT2	
36	CAF	MKRUPTBB		
37	XCH	BBANK		
38	TCF	MARKRUPT		
39				
40	DXCH	ARUPT	# UPRUPT	
41	CAF	UPRPTBB		
42	XCH	BBANK		
43	TCF	UPRUPT		
44				
45	DXCH	ARUPT	# DOWNRUPT	
46	CAF	DWNRPTBB		
47	XCH	BBANK		
48	TCF	DODOWNTM		
49				
50	DXCH	ARUPT	# RADAR RUPT	
51	CAF	RDRPTBB		

[illegible]

SWITCHED-BANK PORTION

BANK 12
SETLOC T4RUP
BANK

CDRVE

COUNT* \$\$/T4RPT
CCS DSPTAB +11D
TC DSPOUT
TC DSPOUT

XCH DSPTAB +11D
MASK LOW11
TS DSPTAB +11D
AD RELTAB11
EXTEND
WRITE OUT0
TC HANG20

# DSPOUT PROGRAM, PUTS OUT DISPLAYS			
DSPOUTSB	TS	NOUT	
	CS	ZERO	
	TS	DSRUPTM	# SET TO -0 FOR 1ST PASS THRU DSPTAB
	XCH	DSPCNT	
	AD	NEG0	# TO PREVENT +0
	TS	DSPCNT	
DSPSCAN	INDEX	DSPCNT	
	CCS	DSPTAB	
	CCS	DSPCNT	# IF DSPTAB ENTRY +, SKIP
	TCF	DSPSCAN -2	# IF DSPCNT +, TRY AGAIN
	TCF	DSPLAY	# IF DSPTAB ENTRY -, DISPLAY
TABLNTH	OCT	12	# DEC 10, LENGTH OF DSPTAB
	CCS	DSRUPTM	# IF DSRUPTM=+0, 2ND PASS THRU DSPTAB
120MRUPT	DEC	16372	# (DSPCNT = 0). +0 INTO NOUT.
	TS	NOUT	
	TC	Q	
	TS	DSRUPTM	# IF DSRUPTM=-0, 1ST PASS THRU DSPTAB
	CAF	TABLNTH	# (DSPCNT=0).+0 INTO DSRUPTM. PASS AGAIN
	TCF	DSPSCAN -1	
DSPLAY	AD	ONE	
	INDEX	DSPCNT	
	TS	DSPTAB	# REPLACE POSITIVELY
	MASK	LOW11	# REMOVE BITS 12 TO 15
	TS	DSRUPTM	
	CAF	HI5	
	INDEX	DSPCNT	
	MASK	RELTAB	# PICK UP BITS 12 TO 15 OF RELTAB ENTRY
	AD	DSRUPTM	
	EXTEND		
	WRITE	OUT0	
	TCF	Q+1	
DSPOUT	CCS	FLAGWRD5	# IS DSKY FLAG ON
	CAF	ZERO	# NO
	TCF	NODSPOUT	# NO
	CCS	NOUT	# YES
	TC	DSPOUTSB	
	TCF	NODSPOUT	# NO DISPLAY REQUESTS
HANG20	CS	14,11,9	
	ADS	DSRUPTSW	
	CAF	20MRUPT	
SETTIME4	TS	TIME4	

1	# THE STATUS OF THE PROCEED PUSHBUTTON IS MONITORED EVERY 120 MILLISECONDS VIA THE CHANNEL 32 BIT 14 INBIT.				1
2	# THE STATE OF THIS INBIT IS COMPARED WITH ITS STATE DURING THE PREVIOUS T4RUPT AND IS PROCESSED AS FOLLOWS.				2
3					3
4	# IF PREV ON AND NOW ON -- BYPASS.				4
5	# IF PREV ON AND NOW OFF -- UPDATE IMODES33.				5
6	# IF PREV OFF AND NOW ON -- UPDATE IMODES33 AND PROCESS VIA PINBALL.				6
7	# IF PREV OFF AND NOW OFF -- BYPASS.				7
8	# THE LOGIC EMPLOYED REQUIRES ONLY 9 MCT (APPROX. 108 MICROSECONDS) OF COMPUTER TIME WHEN NO CHANGES OCCUR.				8
9					9
10	PROCEEDE	CA	IMODES33	# MONITOR FOR PROCEED BUTTON	10
11		EXTEND			11
12		RXOR	CHAN32		12
13		MASK	BIT14		13
14		EXTEND			14
15		BZF	T4JUMP	# NO CHANGE	15
16					16
17		LXCH	IMODES33		17
18		EXTEND			18
19		RXOR	LCHAN		19
20		TS	IMODES33	# UPDATE IMODES33	20
21		MASK	BIT14		21
22		CCS	A		22
23		TCF	T4JUMP	# WAS ON -- NOW OFF	23
24					24
25		CAF	CHRPRI0	# WAS OFF -- NOW ON	25
26		TC	NOVAC		26
27		EBANK=	DSPCOUNT		27
28		2CADR	PROCKEY		28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60



1	# JUMP TO APPROPRIATE ONCE-PER SECOND (0.96 SEC ACTUALLY) ACTIVITY				1
2					2
3					3
4	T4JUMP	INDEX	RUPTREG1		4
5		TCF	+1		5
6					6
7		TC	RCSMONIT		7
8		TCF	RRAUTCHK		8
9		TCF	IMUMON		9
10		TCF	DAPT4S		10
11		TC	RCSMONIT		11
12		TCF	RRAUTCHK		12
13		TCF	IMUMON		13
14		TCF	DAPT4S		14
15					15
16	20MRUPT	=	OCT37776	# (DEC 16382)	16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

ADDITIONAL ROUTINES FOR 20MS. KEYBOARD ACTIVITY

NODSPOUT

EXTEND
WRITE OUT0CAF 120MRUPT #SET FOR NEXT CCRIVE
TCF SETTIME4

QUIKDSP

CAF BIT14
MASK DSRUPTSW
EXTEND

BZF QUIKOFF # WROTE LAST TIME, NOW TURN OFF RELAYS.

CCS NOUT

TC DSPOUTSB
TCF NODSPY # NOUT=0 OR BAD RETURN FROM DSPOUTSB
CS BIT14 # GOOD RETURN (WE DISPLAYED SOMETHING)

QUIKRUPT

ADS DSRUPTSW

CAF 20MRUPT
TS TIME4CAF BIT9
ADS DSRUPTSW

TC RESUME

NODSPY

EXTEND
WRITE OUT0

SYNCT4

CAF 20MRUPT
ADS TIME4CAF BIT9
ADS DSRUPTSWCCS DSRUPTSW
TC RESUME

OCT37737

OCT 37737
TC SYNCT4
TC RESUME

QUIKOFF

EXTEND
WRITE OUT0CAF BIT14 # RESET DSRUPTSW TO SEND DISPLAY NEXT PASS
TCF QUIKRUPT

14,11,9

OCT 22400

PROGRAM NAME: IMUMON

FUNCTIONAL DESCRIPTION: THIS PROGRAM IS ENTERED EVERY 480 MS. IT DETECTS CHANGES OF THE IMU STATUS BITS IN
CHANNEL 30 AND CALLS THE APPROPRIATE SUBROUTINES. THE BITS PROCESSED AND THEIR RELEVANT SUBROUTINES ARE:

#	FUNCTION	BIT	SUBROUTINE CALLED
#	-----	---	-----
#	TEMP IN LIMITS	15	TLIM
#	ISS TURN-ON REQUEST	14	ITURNON
#	IMU FAIL	13	IMUFAIL (SETISSW)
#	IMU CDU FAIL	12	ICDUFAIL (SETISSW)
#	IMU CAGE	11	IMUCAGE
#	IMU OPERATE	9	IMUOP

THE LAST SAMPLED STATE OF THESE BITS IS LEFT IN IMODES30. ALSO, EACH SUBROUTINE CALLED FINDS THE NEW
VALUE OF THE BIT IN A, WITH Q SET TO THE PROPER RETURN LOCATION NXTIFAIL.

CALLING SEQUENCE: T4RUPT EVERY 480 MILLISECONDS.

JOBS OR TASKS INITIATED: NONE.

SUBROUTINES CALLED: TLIM, TURNON, SETISSW, IMUCAGE, IMUOP.

ERASABELE INITIALIZATION:

FRESH START OR RESTART WITH NO GROUPS ACTIVE: C((MODES30) = OCT 37411).

RESTART WITH ACTIVE GROUPS: C(IMODES30) = (B(IMODES30)AND(OCT 00035)) PLUS OCT 37400.
THIS LEAVES IMU FAIL BITS INTACT.

ALARMS: NONE.

EXIT: TNONTEST.

OUTPUT: UPDATED IMODES30 WITH CHANGES PROCESSED BY APPROPRIATE SUBROUTINE.

IMUMON CA IMODES30 # SEE IF THERE HAS BEEN A CHANGE IN THE
EXTEND # RELEVANT BITS OF CHAN 30.

RXOR CHAN30
MASK 30RDMSK

EXTEND
BZF TNONTEST # NO CHANGE IN STATUS

TS RUPTREG1 # SAVE BITS WHICH HAVE CHANGED.
LXCH IMODES30 # UPDATE IMODES30.

EXTEND
RXOR LCHAN
TS IMODES30

CS ONE
XCH RUPTREG1
EXTEND

21	THE
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
80	

PROGRAM NAME: TNONTEST.

FUNCTIONAL DESCRIPTION: THIS PROGRAM HONORS REQUESTS FOR ISS INITIALIZATION. ISS TURN-ON (CHANNEL 30 BIT 14)
AND ISS OPERATE (CHANNEL 30 BIT 9) REQUESTS ARE TREATED AS A PAIR AND PROCESSING TAKES PLACE .480 SECONDS
AFTER EITHER ONE APPEARS. THIS INITIALIZATION TAKES ON ONE OF THE FOLLOWING THREE FORMS:

1) ISS TURN-ON: IN THIS SITUATION THE COMPUTER IS OPERATING WHEN THE ISS IS TURNED ON. NOMINALLY,
BOTH ISS TURN-ON AND ISS OPERATE APPEAR. THE PLATFORM IS CAGED FOR 90 SECONDS AND THE ICDU'S ZEROED
SO THAT AT THE END OF THE PROCESS THE GIMBAL LOCK MONITOR WILL FUNCTION PROPERLY.

2) ICDU INITIALIZATION: IN THIS CASE THE COMPUTER WAS PROBABLY TURNED ON WITH THE ISS IN OPERATE OR
A FRESH START WAS DONE WITH THE ISS IN OPERATE. IN THIS CASE ONLY ISS OPERATE IS ON. THE ICDU'S ARE
ZEROED SO THE GIMBAL LOCK MONITOR WILL FUNCTION. AN EXCEPTION IS IF THE ISS IS IN GIMBAL LOCK AFTER
A RESTART, THE ICDU'S WILL NOT BE ZEROED.

3) RESTART WITH RESTARTABLE PROGRAM USING THE IMU: IN THIS CASE, NO INITIALIZATION TAKES PLACE SINCE
IT IS ASSUMED THAT THE USING PROGRAM DID THE INITIALIZATION AND THEREFORE T4RUPT SHOULD NOT INTERFERE.

IMODES30 BIT 7 IS SET = 1 BY THE FIRST BIT (CHANNEL 30 BIT 14 OR 9) WHICH ARRIVES. FOLLOWING THIS, TNONTEST IS
ENTERED, FINDS BIT 7 = 1 BUT BIT 8 = 0, SO IT SETS BIT 8 = 1 AND EXITS. THE NEXT TIME IT FINDS BIT 8 = 1 AND
PROCEEDS, SETTING BITS 8 AND 7 = 0. AT PROCTNON, IF ISS TURN-ON REQUEST IS PRESENT, THE ISS IS CAGED (ZERO +
COARSE). IF ISS OPERATE IS NOT PRESENT PROGRAM ALARM 00213 IS ISSUED. AT THE END OF A 90 SECOND CAGE, BIT 2
OF IMODES30 IS TESTED. IF IT IS = 1, ISS TURN-ON WAS NOT PRESENT FOR THE ENTIRE 90 SECONDS. IN THAT CASE, IF
THE ISS TURN-ON REQUEST IS PRESENT THE 90 SECOND WAIT IS REPEATED. OTHERWISE NO ACTION OCCURS UNLESS A PROGRAM
WAS WAITING FOR THE INITIALIZATION IN WHICH CASE THE PROGRAM IS GIVEN AN IMUSTALL ERROR RETURN. IF THE DELAY
WENT PROPERLY, THE ISS DELAY OUTBIT IS SENT AND THE ICDU'S ZEROED. A TASK IS INITIATED TO REMOVE THE PIPA FAIL
INHIBIT BIT IN 10.24 SECONDS. IF A MISSION PROGRAM WAS WAITING IT IS INFORMED VIA ENDIMU.

AT PROCTNON, IF ONLY ISS OPERATE IS PRESENT (OPONLY), THE CDU'S ARE ZEROED UNLESS THE PLATFORM IS IN COARSE
ALIGN (= GIMBAL LOCK HERE) OR A MISSION PROGRAM IS USING THE IMU (INUSEFLG = 1).

CALLING SEQUENCE: T4RUPT EVERY 480 MILLISECONDS AFTER IMUMON.

JOBS OR TASKS INITIATED: 1) ENDTNON, 90 SECONDS AFTER CAGING STARTED. 2) ISSUP, 4 SECONDS AFTER CAGING DONE.
3) PFAILOK, 10.24 SECONDS AFTER INITIALIZATION COMPLETED. 4) UNZ2, 320 MILLISECONDS AFTER ZEROING
STARTED.

SUBROUTINES CALLED: CAGESUB, CAGESUB2, ZEROICDU, ENDIMU, IMUBAD, NOATTOFF, SETISSW, VARDELAY.

ERASABLE INITIALIZATION: SEE IMUMON.

ALARMS: PROGRAM ALARM 00213 IF ISS TURN-ON REQUESTED WITHOUT ISS OPERATE.

EXIT: ENDTNON EXITS TO C33TEST. TASKS HAVING TO DO WITH INITIALIZATION EXIT AS FOLLOWS: MISSION PROGRAM
WAITING AND INITIALIZATION COMPLETE, EXIT TO ENDIMU, MISSION PROGRAM WAITING AND INITIALIZATION FAILED, EXIT TO
IMUBAD, IMU NOT IN USE, EXIT TO TASKOVER.

OUTPUT: ISS INITIALIZED.

TNONTEST CS IMODES30 # AFTER PROCESSING ALL CHANGES, SEE IF IT

```
1
2      MASK      BIT7      # IS TIME TO ACT ON A TURN-ON SEQUENCE.
3      CCS      A
4      TCF      C33TEST    # NO -- EXAMINE CHANNEL 33.
5
6      CAF      BIT8      # SEE IF FIRST SAMPLE OR SECOND.
7      MASK      IMODES30
8      CCS      A
9      TCF      PROCTNON    # REACT AFTER A SECOND SAMPLE.
10
11     CAF      BIT8      # IF FIRST SAMPLE, SET BIT TO REACT NEXT
12     ADS      IMODES30    # TIME.
13     TCF      C33TEST
14
15     # PROCESS IMU TURN-ON REQUESTS AFTER WAITING 1 SAMPLE FOR ALL SIGNALS TO ARRIVE.
16
17     PROCTNON      CS      BITS7&8
18                  MASK    IMODES30
19                  TS      IMODES30
20                  MASK    BIT14      # SEE IF TURN-ON REQUEST.
21                  CCS      A
22                  TCF      OPONLY    # OPERATE ON ONLY.
23
24                  CS      IMODES30    # IF TURN-ON REQUEST, WE SHOUD HAVE IMU
25                  MASK    BIT9      # OPERATE.
26                  CCS      A
27                  TCF      +3
28
29                  TC      ALARM      # ALARM IF NOT
30                  OCT      213
31
32     +3            TC      CAGESUB
33                  CAF      90SECS
34                  TC      WAITLIST
35                  EBANK=   M11
36                  2CADR    ENDTNON
37
38                  TCF      C33TEST
39
40     RETNON        CAF      90SECS
41                  TC      VARDELAY
42
43     ENDTNON        CS      BIT2      # RESET TURN-ON REQUEST FAIL BIT.
44                  MASK    IMODES30
45                  XCH      IMODES30
46                  MASK    BIT2      # IF IT WAS OFF, SEND ISS DELAY COMPLETE.
47                  EXTEND
48                  BZF      ENDTNON2
49
50                  CAF      BIT14      # IF IT WAS ON AND TURN-ON REQUEST NOW.
51
52
53
54
55
56
57
58
59
60
```

	MASK	IMODES30	# PRESENT, RE-ENTER 90 SEC DELAY IN WL.
	EXTEND		
	BZF	RETNON	
	CS	FLAGWRD0	# IF IT IS NOT ON NOW, SEE IF A PROG WAS
	MASK	IMUSEBIT	# WAITING.
	CCS	A	
	TCF	TASKOVER	
	TC	POSTJUMP	
	CADR	IMUBAD	# UNSUCCESSFUL TURN-ON.
ENDTNON2	CAF	BIT15	# SEND ISS DELAY COMPLETE.
	EXTEND		
	WOR	CHAN12	
	TC	IBNKCALL	# TURN OFF NO ATT LAMP.
	CADR	NOATTOFF	
UNZ2	TC	ZEROICDU	
	CS	BITS4&5	# REMOVE ZERO AND COARSE.
	EXTEND		
	WAND	CHAN12	
	CAF	BIT11	# WAIT 10 SECS FOR CTRS TO FIND GIMBALS
	TC	VARDELAY	
ISSUP	CS	OCT54	# REMOVE CAGING, IMU FAIL INHIBIT BIT, AND
	MASK	IMODES30	# ICDUFAIL INHIBIT FLAGS.
	TS	IMODES30	
	CS	BIT6	# ENABLE DAP
	MASK	IMODES33	
	TS	IMODES33	
	CS	FLAGWRD2	# TEST DRIFTFLG: IF ON DO NOTHING BECAUSE
	MASK	DRFTBIT	# IMUCOMP SHOUD BE ALL SET UP (RESTART
	EXTEND		# WITH IMUSE DOWN). IF OFF, SET DRIFTFLG
	BZF	+4	# AND 1/PIPADT TO GET FREEFALL IMUCOMP
	ADS	FLAGWRD2	# GOING (FRESH START OR ISS TURN-ON).
	CA	TIME1	
	XCH	1/PIPADT	# CANNOT GET HERE IF RESTART WITH IMUSE UP
	TC	SETISSW	# ISS WARNING MIGHT HAVE BEEN INHIBITED.
	CS	BIT15	# REMOVE IMU DELAY COMPLETE DISCRETE.
	EXTEND		
	WAND	CHAN12	
	CAF	4SECS	# DON'T ENABLE PROG ALARM ON PIP FAIL FOR

1					1
2		TC	WAITLIST	# ANOTHER 4 SECS.	2
3		EBANK=	CDUIND		3
4		2CADR	PFAILOK		4
5					5
6		TCF	TASKOVER		6
7					7
8	OPONLY	CAF	BIT4	# IF OPERATE ON ONLY, AND WE ARE IN COARSE	8
9		EXTEND		# ALIGN, DON'T ZERO THE CDUS BECAUSE WE	9
10		RAND	CHAN12	# MIGHT BE IN GIMBAL LOCK.	10
11		CCS	A		11
12		TCF	C33TEST		12
13					13
14		CAF	IMUSEBIT	# OTHERWISE, ZERO THE COUNTERS.	14
15		MASK	FLAGWRDO	# UNLESS SOMEONE IS USING TH IMU.	15
16		CCS	A		16
17		TCF	C33TEST		17
18					18
19		TC	CAGESUB2	# SET TURNON FLAGS.	19
20					20
21	ISSZERO	TC	IBNKCALL	# TURN OFF NO ATT LAMP.	21
22		CADR	NOATTOFF	# IMU CAGE OFF ENTRY.	22
23					23
24		CAF	BIT5	# ISS CDU ZERO	24
25		EXTEND			25
26		WOR	CHAN12		26
27					27
28		TC	ZEROICDU		28
29		CAF	BIT6	# WAIT 300 MS. FOR AGS TO RECEIVE SIGNAL.	29
30		TC	WAITLIST		30
31		EBANK=	M11		31
32		2CADR	UNZ2		32
33					33
34		TCF	C33TEST		34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

PROGRAM NAME: C33TEST

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM MONITORS THREE FLIP-FLOP INBITS OF CHANNEL 33 AND CALLS THE APPROPRIATE
SUBROUTINE TO PROCESS A CHANGE. IT IS ANALOGOUS TO IMUMON, WHICH MONITORS CHANNEL 30, EXCEPT THAT IT READS
CHANNEL 33 WITH A WAND INSTRUCTION BECAUSE A 'WRITE' PULSE IS REQUIRED TO RESET THE FLIP-FLOPS. THE BITS

PROCESSED AND THE SUBROUTINES CALLED ARE:

BIT	FUNCTION	SUBROUTINE
13	PIPA FAIL	PIPFAIL
12	DOWNLINK TOO FAST	DNTMFAST
11	UPLINK TOO FAST	UPTMFAST

UPON ENTRY TO THE SUBROUTINE, THE NEW BIT STATE IS IN A.

CALLING SEQUENCE: EVERY 480 MILLISECONDS AFTER TNONTEST.

JOBS OR TASKS INITIATED: NONE.

SUBROUTINES CALLED: PIPFAIL, DNTMFAST AND UPTMFAST ON BIT CHANGES.

ERASABLE INITIALIZATION: C(IMODES33) = OCT 16000 ON A FRESH START OR RESTART, THEREFORE, THESE ALARMS WILL
REAPPEAR IF THE CONDITIONS PERSIST.

ALARMS: NONE.

EXIT: GLOCKMON.

OUTPUT: UPDATED BITS 13, 12, AND 11 OF IMODES33 WITH CHANGES PROCESSED.

C33TEST	CA	IMODES33	# SEE IF RELEVANT CHAN33 BITS HAVE
	MASK	33RDMSK	
	TS	L	# CHANGED.
	CAF	33RDMSK	
	EXTEND		
	WAND	CHAN33	# RESETS FLIP-FLOP INPUTS
	EXTEND		
	RXOR	LCHAN	
	EXTEND		
	BZF	GLOCKMON	# ON NO CHANGE.
	TS	RUPTREG1	# SAVE BITS WHICH HAVE CHANGED.
	LXCH	IMODES33	
	EXTEND		
	RXOR	LCHAN	
	TS	IMODES33	# UPDATED IMODES33.
	CAF	ZERO	
	XCH	RUPTREG1	
	DOUBLE		

142THE

PROGRAM NAME: GLOCKMON

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM MONITORS THE CDUZ COUNTER TO DETERMINE WHETHER THE ISS IS IN GIMBAL LOCK
AND TAKES ACTION IF IT IS. THREE REGIONS OF MIDDLE GIMBAL ANGLE (MGA) ARE USED:

#

1) ABS(MGA) LESS THAN OR EQUAL TO 70 DEGREES -- NORMAL MODE.

2) ABS(MGA) GREATER THAN 70 DEGREES AND LESS THAN OR EQUAL TO 85 DEGREES -- GIMBAL LOCK LAMP TURNED ON.

3) ABS(MGA) GREATER THAN 85 DEGREES -- ISS PUT IN COARSE ALIGN AND NO ATT LAMP TURNED ON.

#

CALLING SEQUENCE: EVERY 480 MILLISECONDS AFTER C33TEST.

#

JOBS OR TASKS INITIATED: NONE.

#

SUBROUTINES CALLED: 1) SETCOARS WHEN ABS(MGA) GREATER THEN 85 DEGREES AND ISS NOT IN COARSE ALIGN.

2) LAMPTST BEFORE TURNING OFF GIMBAL LOCK LAMP.

#

ERASABLE INITIALIZATION:

1) FRESH START OR RESTART WITH NO GROUPS ACTIVE: C(CDUZ) = 0, IMODES30 BIT 6 = 0, IMODES33 BIT 1 = 0.

2) RESTART WITH GROUPS ACTIVE: SAME AS FRESH START EXCEPT C(CDUZ) NOT CHANGED SO GIMBAL MONITOR
PROCEEDS AS BEFORE.

#

ALARMS: 1) MGA REGION (2) CAUSES GIMBAL LOCK LAMP TO BE LIT.

2) MGA REGION (3) CAUSES THE ISS TO BE PUT IN COARSE ALIGN AND THE NO ATT LAMP TO BE LIT IF EITHER NOT
SO ALREADY.

#

GLOCKMON CCS CDUZ

TCF GLOCKCHK # SEE IF MAGNITUDE OF MGA IS GREATER THAN
TCF SETGLOCK # 70 DEGREES.

TCF GLOCKCHK

TCF SETGLOCK

#

GLOCKCHK AD -70DEGS

EXTEND

BZMF SETGLOCK -1 # NO LOCK.

#

AD -15DEGS # SEE IF ABS(MGA) GREATER THAN 85 DEGREES

EXTEND

BZMF NOGIMRUN

#

CAF BIT4 # IF SO, SYSTEM SHOULD BE IN COARSE ALIGN
EXTEND # TO PREVENT GIMBAL RUNAWAY.

RAND CHAN12

CCS A

TCF NOGIMRUN

#

TC IBNKCALL

CADR SETCOARS

#

CAF SIX # ENABLE ISS ERROR COUNTERS IN 60 MS.

TC WAITLIST

#

#

#

#

#

#

#

#

#

	EBANK=	CDUIND	
	2CADR	CA+ECE	
NOGIMRUN	CAF	BIT6	# TURN ON GIMBAL LOCK LAMP.
	TCF	SETGLOCK	
-1	CAF	ZERO	
SETGLOCK	AD	DSPTAB +11D	# SEE IF PRESENT STATE OF GIMBAL LOCK LAMP
	MASK	BIT6	# AGREES WITH DESIRED STATE BY HALF ADDING
	EXTEND		# THE TWO.
	BZF	GLOCKOK	# OK AS IS.
	MASK	DSPTAB +11D	# IF OFF, DON'T TURN ON IF IMU BEING CAGED.
	CCS	A	
	TCF	GLAMPTST	# TURN OFF UNLESS LAMP TEST IN PROGRESS.
	CAF	BIT6	
	MASK	IMODES30	
	CCS	A	
	TCF	GLOCKOK	
GLINVERT	CS	DSPTAB +11D	# INVERT GIMBAL LOCK LAMP.
	MASK	BIT6	
	AD	BIT15	# TO INDICATE CHANGE IN DSPTAB +11D.
	XCH	DSPTAB +11D	
	MASK	OCT37737	
	ADS	DSPTAB +11D	
	TCF	GLOCKOK	
GLAMPTST	TC	LAMPTEST	# TURN OFF UNLESS LAMP TEST IN PROGRESS.
	TCF	GLOCKOK	
	TCF	GLINVERT	
-70DEGS	DEC	-.38888	# -70 DEGREES SCALED IN HALF-REVOLUTIONS.
-15DEGS	DEC	-.08333	

PROGRAM NAME: TLIM.
#

FUNCTIONAL DESCRIPTION: THIS PROGRAM MAINTAINS THE TEMP LAMP (BIT 4 OF CHANNEL 11) ON THE DSKY TO AGREE WITH
THE TEMP SIGNAL FROM THE ISS (BIT 15 OF CHANNEL 30). HOWEVER, THE LIGHT WILL NOT BE TURNED OFF IF A LAMP TEST
IS IN PROGRESS.

CALLING SEQUENCE: CALLED BY IMUMON ON A CHANGE OF BIT 15 OF CHANNEL 30.
#

JOBS OR TASKS INITIATED: NON.

SUBROUTINES CALLED: LAMPTEST.
#

ERASABLE INITIALIZATION: FRESH START AND RESTART TURN THE TEMP LAMP OFF.
#

ALARMS: TEMP LAMP TURNED ON WHEN THE IMU TEMP GOES OUT OF LIMITS.

EXIT: NXTIFAIL.
#

OUTPUT: SERVICE OF TEMP LAMP. IN A, EXCEPT FOR TLIM.

TLIM	MASK	POSMAX	# REMOVE BIT FROM WORD OF CHANGES AND SET
	TS	RUPTREG2	# DSKY TEMP LAMP ACCORDINGLY.

CCS	IMODES30
TCF	TEMPOK
TCF	TEMPOK

CAF	BIT4	# TURN ON LAMP.
EXTEND		

WOR	DSALMOUT
TCF	NXTIFAIL

TEMPOK	TC	LAMPTEST	# IF TEMP NOW OK, DON'T TURN OFF LAMP IF
	TCF	NXTIFAIL	# LAMP TEST IN PROGRESS.

CS	BIT4	# TURN OFF LAMP
EXTEND		
WAND	DSALMOUT	
TCF	NXTIFAIL	

PROGRAM NAME: ITURNON.

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM IS CALLED BY IMUMON WHEN A CHANGE OF BIT 14 OF CHANNEL 30 (ISS TURN-ON REQUEST) IS DETECTED. UPON ENTRY, ITURNON CHECKS IF A TURN-ON DELAY SEQUENCE HAS FAILED, AND IF SO, IT EXITS. IF NOT, IT CHECKS WHETHER THE TURN-ON REQUEST CHANGE IS TO ON OR OFF. IF ON, IT SETS BIT7 OF IMODES30 TO 1 SO THAT TNONTEST WILL INITIATE THE ISS INITIALIZATION SEQUENCE. IF OFF, THE TURN-ON DELAY SIGNAL, CHANNEL 12 BIT 15, IS CHECKED AND IF IT IS ON, ITURNON EXITS. IF THE DEALY SIGNAL IS OFF, PROGRAM ALARM 00207 IS ISSUED, BIT 2 OF IMODES30 IS SET TO 1 AND THE PROGRAM EXITS.

#

THE SETTING OF BIT 2 OF IMODES30 (ISS DELAY SEQUENCE FAIL) INHIBITS THIS ROUTINE AND IMUOP FROM PROCESSING ANY CHANGES. THIS BIT WILL BE RESET BY THE ENDTNOM ROUTINE WHEN THE CURRENT 90 SECOND DELAY PERIOD ENDS.

#

CALLING SEQUENCE: FROM IMUMON WHEN ISS TURN-ON REQUEST CHANGES STATE.

#

JOBS OR TASKS INITIATED: NONE.

#

SUBROUTINES CALLED: ALARM, IF THE ISS TURN-ON REQUEST IS NOT PRESENT FOR 90 SECONDS.

#

ERASABLE INITIALIZATION: FRESH START AND RESTART SET BIT 15 OF CHANNEL 12 AND BITS 2 AND 7 OF IMODES30 TO 0, AND BIT 14 OF IMODES30 TO 1.

#

ALARMS: PROGRAM ALARM 00207 IS ISSUED IF THE ISS TURN-ON REQUEST SIGNAL IS NOT PRESENT FOR 90 SECONDS.

#

EXIT: NXTIFAIL.

#

OUTPUT: BIT 7 OF IMODES30 TO START ISS INITIALIZATION, OR BIT 2 OF IMODES30 AND PROGRAM ALARM 00207 TO INDICATE A FAILED TURN-ON SEQUENCE.

ITURNON

CAF

BIT2

IF DELAY REQUEST HAS GONE OFF

MASK

IMODES30

PREMATURELY, DO NOT PROCESS ANY CHANGES

CCS

A

UNTIL THE CURRENT 90 SEC WAIT EXPIRES.

TCF

NXTIFAIL

CAF

BIT14

SEE IF JUST ON OR OFF.

MASK

IMODES30

EXTEND

BZF

ITURNON2

IF JUST ON.

CAF

BIT15

SEE IF DELAY PRESENT DISCRETE HAS BEEN

RAND

CHAN12

SENT. IF SO, ACTION COMPLETE

EXTEND

BZF

+2

TCF

NXTIFAIL

CAF

BIT2

IF NOT, SET BIT TO INDICATE REQUEST NOT

ADS

IMODES30

PRESENT FOR FULL DURATION.

TC

ALARM

OCT

207

TCF

NXTIFAIL



1				1
2	ITURNON2	CS	IMODES30	2
3		MASK	BIT7	3
4		ADS	IMODES30	4
5		CAF	RRINIT	5
6		TS	RADMODES	6
7		TCF	NXTIFAIL	7
8				8
9	RRINIT	OCT	00102	9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

PROGRAM NAME: IMUCAGE.

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM PROCESSES CHANGES OF THE IMUCAGE INBIT, CHANNEL 30 BITS 11. IF THE BIT
CHANGES TO 0 (CAGE BUTTON PRESSED), THE ISS IS CAGED (ICDU ZERO + COARSE ALIGN + NO ATT LAMP) UNTIL THE
ASTRONAUT SELECTS ANOTHER PROGRAM TO ALIGN THE ISS. ANY PULSE TRAINS TO THE ICDU'S AND GYRO'S ARE TERMINATED,
THE ASSOCIATE OUTCOUNTERS ARE ZEROED AND THE GYRO'S ARE DE-SELECTED. NO ACTION OCCURS WHEN THE BUTTON IS
RELEASED (INBIT CHANGES TO 1).

#

CALLING SEQUENCE: BY IMUMON WHEN IMU CAGE BIT CHANGES.

#

JOBS OR TASKS INITIATED: NONE.

#

SUBROUTINES CALLED: CAGESUB.

#

ERASABLE INITIALZATION: FRESH START AND RESTART SET BIT 11 OF IMODES30 TO 1.

#

ALARMS: NONE.

#

EXIT: NXTIFAIL.

#

OUTPUT: ISS CAGED, COUNTERS ZEROED, PULSE TRAINS TERMINATED AND NO ATT LAMP LIT.

IMUCAGE

CCS

A

NO ACTION OF GOING OFF.

TCF

ISSZERO

CS

OCT77000

TERMINATE ICDU, RCDU, GYRO PULSE TRAINS

EXTEND

WAND

CHAN14

CS

OCT272

KNOCK DOWN DISPLAY INERTIAL DATA, IMU

EXTEND

WAND

CHAN12

ERROR COUNTER ENABLE, ZERO ICDU, COARSE
ALIGN ENABLE, RR ERROR COUNTER ENABLE.

CS

ENGONBIT

INSURE ENGONFLG IS CLEAR.

MASK

FLAGWRD5

TS

FLAGWRD5

CS

PRIO30

TURN ENGINE OFF.

EXTEND

RAND

DSALMOUT

AD

BIT14

EXTEND

WRITE

DSALMOUT

FORCE BIT14=1, BIT13=0.

TC

CAGESUB1

TC

IBNKCALL

KNOCK DOWN TRACK, REFSMMAT, DRIFT FLAGS

CADR

RNDREFDR

CS

ZERO

TS

CDUXCMD

TS

CDUYCMD

TS	CDUZCMD
TS	GYROCMD

CS OCT740

EXTEND

WAND	CHAN14
TCF	NXTIFAIL

```
# HAVING WAITED AT LEAST 27 MCT FROM
# GYRO PULSE TRAIN TERMINATION, WE CAN
# DE-SELECT THE GYROS.
```

PROGRAM NAME: IMUOP.

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM PROCESSES CHANGES IN THE ISS OPERATE DISCRETE, BIT 9 OF CHANNEL 30.

IF THE INBIT CHANGES TO 0, INDICATING ISS ON, IMUOP GENERALLY SETS BIT 7 OF IMODES30 TO 1 TO REQUEST ISS

INITIALIZATION VIA TNONTEST. AN EXCEPTION IS DURING A FAILED ISS DELAY DURING WHICH BIT 2 OF IMODES30 IS SET

TO 1 AND NO FURTHER INITIALIZATION IS REQUIRED. WHEN THE INBIT CHANGES TO 1, INDICATING ISS OFF, IMUSEFLG IS

TESTED TO SEE IF ANY PROGRAM WAS USING THE ISS. IF SO, PROGRAM ALARM 00214 IS ISSUED.

#

CALLING SEQUENCE: BY IMUMON WHEN BIT 9 OF CHANNEL 30 CHANGES.

#

JOBS OR TAKS INITIATED: NONE.

#

SUBROUTINES CALLED: ALARM, IF ISS IS TURNED OFF WHILE IN USE.

#

ERASABLE INITIALIZATION: ON FRESH START AND RESTART, BIT 9 OF IMODES30 IS SET TO 1 EXCEPT WHEN THE GIMBAL LOCK

LAMP IS ON, IN WHICH CASE IT IS SET TO 0. THIS PREVENTS ICDU ZERO BY TNONTEST WITH THE ISS IN GIMBAL LOCK.

#

ALARMS: PROGRAM ALARM 00214 IF THE ISS IS TURNED OFF WHILE IN USE.

#

EXIT: NXTIFAIL.

#

OUTPUT: ISS INITIALIZATION REQUEST (IMODES30 BIT 7) OR PROGRAM ALARM 00214.

IMUOP

EXTEND

BZF

IMUOP2

CS

IMODES33

DISABLE DAP

MASK

BIT6

ADS

IMODES33

TC

IBNKCALL

KNOCK DOWN TRACK, REFSMMAT, DRIFT FLAGS

CADR

RNDREFDR

CS

BITS7&8

KNOCK DOWN RENDEZVOUS, IMUUSE FLAGS

MASK

FLAGWRDO

XCH

FLAGWRDO

IF GOING OFF, ALARM IF PROG USING IMU.

COM

MASK

IMUSEFLG

CCS

A

TCF

NXTIFAIL

TC

ALARM

OCT

214

TCF

NXTIFAIL

IMUOP2

CAF

BIT2

SEE IF FAILED ISS TURN-ON SEQ IN PROG.

MASK

IMODES30

CCS

A

TCF

NXTIFAIL

IF SO, DON'T PROCESS UNTIL PRESENT 90

TCF

ITURNON2

SECONDS EXPIRES.

PROGRAM NAME: PIPFAIL

#

FUNCITONAL DESCRIPTION: THIS PROGRAM PROCESSES CHANGES OF BIT 13 OF CHANNEL 33, PIPA FAIL. IT SETS BIT 10 OF IMODES30 TO AGREE. IT CALLS SETISSW IN CASE A PIPA FAIL NECESSITATES AN ISS WARNING. IF NOT, I.E., IMODES30 BIT 1 = 1, AND A PIPA FAIL IS PRESENT AND THE ISS NOT BEING INITIALIZED, PROGRAM ALARM 0212 IS ISSUED.

#

CALLING SEQUENCE: BY C33TEST ON CHANGES OF CHANNEL 33 BIT 13.

#

JOBS OR TASKS INITIATED: NONE.

#

SUBROUTINES CALLED: 1) SETISSW, AND 2) ALARM (SEE FUNCITONAL DESCRIPTION).

#

ERASABLE INITIALZIZATION: SEE IMUMON FOR INITIALIZATION OF IMODES30. THE RELEVANT BITS ARE 5, 7, 8, 9, AND 10.

#

ALARMS: PROGRAM ALARM 00212 IF PIPA FAIL IS PRESENT BUT NEITHER ISS WARNING IS TO BE ISSUED NOR THE ISS IS BEING INITIALIZED.

#

EXIT: NXTFL33.

#

OUTPUT: PROGRAM ALARM 00212 AND ISS WARNING MAINTENANCE.

PIPFAIL

CCS

A

SET BIT10 IN IMODES30 SO ALL ISS WARNING

CAF

BIT10

INFO IS IN ONE REGISTER.

XCH

IMODES30

MASK

-BIT10

ADS

IMODES30

TC

SETISSW

CS

IMODES30

IF PIP FAIL DOESN'T LIGHT ISS WARNING, DO

MASK

BIT1

A PROGRAM ALARM IF IMU OPERATING BUT NOT

CCS

A

CAGED OR BEING TURNED ON.

TCF

NXTFL33

CA

IMODES30

MASK

OCT1720

CCS

A

TCF

NXTFL33

ABOVE CONDITION NOT MET.

TC

ALARM

OCT

212

TCF

NXTFL33

PROGRAM NAMES: DNTMFAST, UPTMFAST
#

FUNCTIONAL DESCRIPTION: THESE PROGRAMS PROCESS CHANGES OF BITS 12 AND 11 OF CHANNEL 33. IF A BIT CHANGES TO A
0, A PROGRAM ALARM IS ISSUED. THE LAARMS ARE:
#

#	BIT	ALARM	CAUSE
#	---	-----	-----
#	12	01105	DOWNLINK TOO FAST
#	11	01106	UPLINK TOO FAST

CALLING SEQUENCE: BY C33TEST ON A BIT CHANGE.

SUBROUTINES CALLED: ALARM, IF A BIT CHANGES TO A 0.
#

ERASABLE INITIALIZATION: FRESH START OR RESTART, BITS 12 AND 11 OF IMODES33 ARE SET TO 1.

ALARMS: SET FUNCTGIONAL DESCRIPTION.

EXIT: NXTFL33.
#

OUTPUT: PROGRAM ALARM ON A BIT CHANGE TO 0.

DNTMFAST CCS A # DO PROG ALARM IF TM TOO FAST.

TCF NXTFL33

TC ALARM

OCT 1105

TCF NXTFL33

UPTMFAST CCS A # SAME AS DNLINK TOO FAST WITH DIFFERENT
TCF NXTFL33 # ALARM CODE.

TC ALARM

OCT 1106

TCF NXTFL33

PROGRAM NAME: SETISSW

#

FUNCTIONAL DESCRIPTION: THIS PROGRAM TURNS THE ISS WARNING LAMP ON AND OFF (CHANNEL 11 BIT 1 = 1 FOR ON,
0 FOR OFF) DEPENDING ON THE STATUS OF IMODES30 BITS 13 (IMU FAIL) AND 4 (INHIBIT IMU FAIL), 12 (ICDU FAIL) AND
3 (INHIBIT ICDU FAIL), AND 10 (PIPA FAIL) AND 1 (INHIBIT PIPA FAIL). THE LAMP IS LEFT ON IF A LAMP TEST IS IN
PROGRESS.

#

CALLING SEQUENCE: CALLED BY IMUMON ON CHANGES TO IMU FAIL AND ICDU FAIL. CALLED BY IFAILOC AND PFAILOC UPON
REMOVAL OF THE FAIL INHIBITS. CALLED BY PIPFAIL WHEN THE PIPA FAIL DISCRETE CHANGES. IT IS CALLED BY PIPUSE
SINCE THE PIPA FAIL PROGRAM ALARM MAY NECESSITATE AN ISS WARNING, AND LIKEWISE BY PIPFREE WHEN THE ALARM DEPARTS
AND IT IS CALLED BY IMUZERO3 AND ISSUP AFTER THE FAIL INHIBITS HAVE BEEN REMOVED.

#

JOBS OR TASKS INITIAZTED: NONE.

#

SUBROUTINES CALLED: NONE.

#

ERASABLE INITIALIZATION:

#

1) IMODES30 -- SEE IMUMON.

2) IMODES33 BIT 1 = 0 (LAMP TEST NOT IN PROGRESS).

#

ALARMS: ISS WARNING.

#

THE FOLLOWING PROGRAM ALARMS WILL SHOW WHICH FAILURE CAUSED THE ISS WARN

PROGRAM ALARM 00777 PIPA FAIL

PROGRAM ALARM 03777 ICDU FAIL

PROGRAM ALARM 04777 ICDU, PIPA FAILS

PROGRAM ALARM 07777 IMU FAIL

PROGRAM ALARM 10777 IMU, PIPA FAILS

PROGRAM ALARM 13777 IMU, ICDU FAILS

PROGRAM ALARM 14777 IMU, ICDU, PIPA FAILS

#

EXIT: VIA Q.

#

OUTPUT: ISS WARNING LAMP SET PROPERLY.

SETISSW

CAF

OCT15

SET ISS WARNING USING THE FAIL BITS IN

MASK IMODES30

BITS 13, 12, AND 10 OF IMODES30 AND THE

EXTEND

FAILURE INHIBIT BITS IN POSITIONS

MP

BIT10

4, 3, AND 1.

CA

IMODES30

EXTEND

ROR

LCHAN

0 INDICATES FAILURE

COM

MASK

OCT15000

CCS

A

TCF

ISSWON

FAILURE.

ISSWOFF

CAF

BIT1

DON'T TURN OFF ISS WARNING IF LAMP TEST

MASK

IMODES33

IN PROGRESS.

[illegible]

JUMP TABLES AND CONSTANTS.

IFAILJMP TCF ITURNON # CHANNEL 30 DISPATCH.

TCF IMUFAIL
TCF ICDUFAIL

30RDMSK TCF IMUCAGE
OCT 76400 # (BIT 10 NOT SAMPLED HERE).
TCF IMUOP

C33JMP TCF PIPFAIL # CHANNEL 33 DISPATCH.

TCF DNTMFAST
TCF UPTMFAST

SUBROUTINE TO SKIP IF LAMP TEST NOT IN PROGRESS.

LAMPTEST CS IMODES33 # BIT 1 OF IMODES33 = 1 IF LAMP TEST IN
MASK BIT1 # PROGRESS.

CCS A
INCR Q
TC Q

33RDMSK EQUALS PRI016
OC40010 OCT 40010
OCT54 OCT 54

OCT75 OCT 75
OCT272 OCT 00272
BITS7&8 OCT 300

OCT1720 OCT 1720
OCT740 OCT 00740
OCT15000 EQUALS PRI015

OCT77000 OCT 77000
BITS6&15 OCT 40040
-BIT10 OCT -1000

90SECS DEC 9000
120MS = OCT14 # (DEC12)

GLOCKOK EQUALS RESUME

PROGRAM NAME: RRAUTCHK
#

FUNCITONAL DESCRIPTION:
RRAUTCHK IS THE RENDEZFOUS RADAR INBIT MONITOR. INITIALLY THE RR
POWER ON AUTO (CHAN 33 BIT 2) INBIT IS CHECKED. IF NO CHANGE, THE
PROGRAM EXITS TO RRCDUCHK. IF A CHANGE, RADMOES IS UPDATED
AND A CHECK MADE IF RR POWER HAS JUST COME ON. IF JUST OFF, A CHECK
IS MADE TO SEE IF A PROGRAM WAS USING THE RR (STATE BIT 7). IF NO,
THE PROGRAM EXITS TO RRCDUCHK. IF YES, PROGRAM ALARM 00514
IS REQUESTED BEFORE EXITING TO RRCDUCHK. IF RR POWER HAS JUST COME
ON, A CHECK IS MADE TO SEE IF A PROGRAM WAS USING THE RR (STATE BIT 7)
SEQUENCE. IF NO, RADMODES IS UPDATED TO INDICATE RR CDU ZERO AND
RR TURN-ON SEQUENCE (BITS 13, 1). A 10 MILLISECOND WAITLIST CALL
IS THEN SET FOR RRTURNON BEFORE THE PROGRAM EXITS TO NORRGMON.

CALLING SEQUENCE:
T4RUPT EVERY 480 MILLISECONDS

ERASABLE INITIALIZATION REQUIRED:
RADMODES, STATE.

SUBROUTINES CALLED:
WAITLIST.

JOBS OR TASKS INITIATED:
RRTURNON

ALARMS: PROGRAM ALARM 00514 -- RADAR GOES OUT OF AUTO MODE WHILE BEING
USED

EXIT: RRCDUCHK, NORRGMON

RRAUTCHK	CA	RADMODES	# SEE IF CHANGE IN RR AUTO MODE BIT.
----------	----	----------	--------------------------------------

EXTEND	
RXOR	CHAN33
MASK	AUTOMBIT
EXTEND	
BZF	RRCDUCHK

LXCH	RADMODES	# UPDATE RADMODES.
------	----------	--------------------

EXTEND

RXOR	LCHAN	# CLR CONT. DES., REMODE, REPOS, CDUZERO,
MASK	OCT05776	# AND TURNON BITS.
TS	RADMODES	# SEE IF JUST ON.

MASK	BIT2	
CCS	A	
TCF	RRCDUCHK -3	# OFF. GO DISABLE RR CDU ERROR COUNTERS.

CA	OCT10001	# SET RRCDUZRO AND TURNON BITS.
ADS	RADMODES	

PROGRAM NAME: RRCDUCHK

#

FUNCTIONAL DESCRIPTION:

RRCDUCHK CHECKS FOR RR CDU FAIL (CHAN 30 BIT 7). INITIALLY THE

RR CDU FAIL BIT IS SAMPLED (CHAN 30 BIT 7). IF NO CHANGE, THE

PROGRAM EXITS TO RRGIMON. IF A CHANGE, THE RR AUTO MODE

(RADMODES BIT 2) BIT IS CHECKED. IF NOT IN RR AUTO MODE, THE

PROGRAM EXITS TO NORRGMOON. IF IN AUTO MODE, RADMODES BIT 7

(RR CDU OK) IS UPDATED AND IF P-20 IS OPERATING PROGRAM ALARM 00515 IS

REQUESTED. CONTROL IS TRANSFERRED TO SETTRKF TO UPDATE

THE TRACKER FAIL LAMP (DSPTAB+11D BIT 8). CONTROL RETURNS TO

RRGIMON.

#

CALLING SEQUENCE:

EVERY 480 MILLISECONDS FROM RRAUTCHK (VIA T4RUPT) UNLESS A

TURN-ON SEQUENCE HAS JUST BE INITIATED.

#

ERASABLE INITIALIZATION REQUIRED:

RADMODES

#

SUBROUTINES CALLED:

SETTRKF

#

JOBS OR TASKS INITIATED:

NONE

#

ALARMS:

TRACKER FAIL

PROGRAM ALARM 00515 -- RRCDU FAIL DURING P-20

#

EXIT:

RRGIMON, NORRGMON

-3

CS BIT2

EXTEND

WAND CHAN12

AT TURNON, DISABLE CDU ERROR COUNTERS.

RRCDUCHK

CA

RADMODES

LAST SAMPLED BIT IN RADMODES.

EXTEND

RXOR CHAN30

MASK RCDUFBIT

EXTEND

BZF RRGIMON

CAF AUTOMBIT

MASK RADMODES

CCS A

TCF NORRGMON

IF RR NOT IN AUTO MODE, DON'T CHANGE BIT
7 OF RADMODES. IF THIS WERE NOT DONE,
THE TRACKER FAIL MIGHT COME ON WHEN
JUST READING LR DATA.

CAF RCDUFBIT

SET BIT 7 OF RADMODES FOR SETTRKF.



1				1
2		LXCH	RADMODES	2
3		EXTEND	# UPDATE RADMODES.	3
4		RXOR	L	4
5		TS	RADMODES	5
6				6
7		CA	RADMODES	7
8		MASK	# DID RR CDU FAIL	8
9		CCS	RCDUFBIT	9
10		TCF	A	10
11		CS	TRKFLCDU	11
12		MASK	# NO	12
13		CCS	FLAGWRDO	13
14		TCF	# RNDVFLG P20 OR P22 OPERATING	14
15		TC	RNDVZBIT	15
16		OCT	A	16
17	TRKFLCDU	TC	TRKFLCDU	17
18			# NO	18
19			# YES	19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

PROGRAM NAME: RRGIMON

#

FUNCTIONAL DESCRIPTION:

RRGIMON IS THE RR GIMBAL LIMIT MONITOR. INITIALLY THE FOLLOWING IS

CHECKED: REMOD, RR CDU'S BEING ZEROED, REPOSITION, AND RR

NOT IN AUTO MODE (RADMODES BITS 14, 13, 11, 2). IF ANY OF THESE

EXIST THE PROGRAM EXITS TO GPMATRIX. IF NONE ARE PRESENT RRLIMCHK

IS CALLED TO SEE IF THE PRESENT RR CDU ANGLES (OPTY, OPTX) ARE WITHIN

THE LIMITS OF THE CURRENT MODE. IF WITHIN LIMITS, THE PROGRAM EXITS

TO NORRGMON. IF NOT WITHIN LIMITS, THE REPOSITION FLAG (RADMODES

BIT 11) IS SET, THE RR AUTO TRACKER AND RR ERROR COUNTER

(CHAN 12 BITS 14, 2) ARE DISABLED, AND A 20 MILLISECOND WAITLIST

CALL IS SET FOR DORREPOS AFTER WHICH THE PROGRAM EXITS TO NORRGMON.

#

CALLING SEQUENCE:

EVERY 480 MILLISECONDS FROM RRCDUCHK (VIA T4RUPT) UNLESS TURN-ON

HAS JUST BEEN INITIATED VIA RRAUTCHK OR IF THERE HAS BEEN A CHANGE IN

THE RR CDU FAIL BIT (CHAN 30 BIT 7) AND THE RR IS NOT IN THE AUTO MODE

(RADMODES BIT 2).

#

ERASABLE INITIALIZATION: RADMODES

#

SUBROUTINES CALLED:

RRLIMCHK, WAITLIST

#

JOBS OR TASKS INITIATED:

DORREPOS

#

ALARMS:

NONE

#

EXIT:

NORRGMON

RRGIMON

CAE

FLAGWRD5

IS NO ANGLE MONITOR FLAG SET

MASK

NORRMBIT

CCS

A

TCF

NORRGMON

YES -- SKIP LIMIT CHECK

CS

FLAGWRD7

IS SERVICER RUNNING?

MASK

AVEGFBIT

CCS

A

TCF

+5

NO. DO R25

CA

FLAGWRD6

YES. IS MUNFLAG SET?

MASK

MUNFLBIT

CCS

A

TCF

NORRGMON

YES. DON'T DO R25

+5

CAF

OCT32002

INHIBIT BY REMODE, ZEROING, MONITOR.

MASK

RADMODES

OR RR NOT IN AUTO.

CCS

A

TCF

NORRGMON

1	# TRACKER PROGRAM			1
2		TC	RRLIMCHK	2
3		ADRES	CDUT	3
4				4
5		TCF	MONREPOS	5
6				6
7		TCF	NORRGMON	7
8				8
9	MONREPOS	CAF	REPOSBIT	9
10		ADS	RADMODES	10
11				11
12		CS	OCT20002	12
13		EXTEND		13
14		WAND	CHAN12	14
15				15
16		CAF	TWO	16
17		TC	WAITLIST	17
18		EBANK=	LOSCOUNT	18
19		2CADR	DORREPOS	19
20				20
21		TCF	NORRGMON	21
22				22
23	OCT32002	OCT	32002	23
24	OCT20002	OCT	20002	24
25	OCT02100	OCT	02100	25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

PROGRAM NAME: GPMATRIX (DAPT4S) MCD. NO. 2 DATE: OCTOBER 27, 1966

#

AUTHOR: JOHNATHAN D. ADDLELSTON (ADAMS ASSOCIATES)

#

MODIFIED: 7FEB. 1968 BY P. S. WEISSMAN TO DELETE COMPUTATION OF MR12 AND MR13, WHICH ARE NO LONGER REQUIRED.

#

THIS PROGRAM CALCULATES ALL THE SINGLE-PRECISION MATRIX ELEMENTS WHICH ARE USED BY LEM DAP TO TRANSFORM VECTORS
FROM GIMBAL TO PILOT (BODY) AXES AND BACK AGAIN. THESE ELEMENTS ARE USED EXCLUSIVELY BY BASIC LANGUAGE ROUTINES
AND THEREFORE ARE NOT ARRAYED FOR USE BY INTERPRETIVE PROGRAMS.

#

CALLING SEQUENCE: GPMATRIX IS TRANSFERRED TO FROM DAPT4S AND IS THUS EXECUTED 4 TIMES A SECOND BY T4RUPT.

DAPT4S IS LISTED IN T4JUMP TABLE TWICE EXPLICITLY AND ALSO OCCURS AFTER RRAUTCHK (WHICH IS ALSO LISTED TWICE).

#

SUBROUTINES CALLED: SPSIN, SPCOS.

#

NORMAL EXIT MODE: TCF RESUME

#

ALARM AND ABORT MODES: NONE.

#

INPUT: CDUX, CDUY, CDUZ.

#

OUTPUT: M11, M21, M32, M22, M32.

#

AOG = CDUX, AIG = CDUY, AMG = CDUZ: MNEMONIC IS : OIM = XYZ

#

#	*	*	SING(MG)	0	1	*
#	M	=	* COS(MG)COS(OG)	SIN(OG)	0	*
#	GP	*	* -COS(MG)SIN(OG)	COS(OG)	0	*

#

#	*	*	0	COS(OG)/COS(MG)	-SIN(OG)/COS(MG)	*
#	M	=	* 0	SIN(OG)	COS(OG)	*
#	PG	*	* 1	-SIN(MG)COS(OG)/COS(MG)	SIN(MG)SIN(OG)/COS(MG)	*

	EBANK=	M11
DAPT4S	EQUALS	GPMATRIX

T4RUPT DAP LOGIC:

GPMATRIX	CAE	CDUZ	# SINGLE ENTRY POINT
	TC	SPSIN	# SIN(CDUZ) = SIN(MG)
	TS	M11	# SCALED AT 1
	CAE	CDUZ	
	TC	SPCOS	# COS(CDUZ) = COS(MG)
	TS	COSMG	# SCALED AT 1 (ONLY A FACTOR)
	CAE	CDUX	
	TC	SPSIN	# SIN(CDUX) = SIN(OG)
	TS	M22	# SCALED AT 1 (ALSO IS MR22)
	CS	M22	

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
80	

PROGRAM DESCRIPTION

AUTHOR: J S MILLER

MODIFIED 6 MARCH 1968 BY P S WEISSMAN TO SET UP JOB FOR 1/ACCS WHEN THE MASKS ARE CHANGED.

THIS ROUTINE IS ATTACHED TO T4RUPT, AND IS ENTERED EVERY 480 MS. ITS FUNCTION IS TO EXAMINE THE LOW 8 BITS OF CHANNEL 32 TO SEE IF ANY ISOLATION-VALVE CLOSURE BITS HAVE APPEARED OR DISAPPEARED (THE CREW IS WARNED OF JET FAILURES BY LAMPS LIT BY THE GRUMMAN FAILURE-DETECTION CIRCUITRY; THEY MAY RESPOND BY OPERATING SWITCHES WHICH ISOLATE PAIRS OF JETS FROM THE PROPELLANT TANKS AND SET BITS IN CHANNEL 32). IN THE EVENT THAT CHANNEL 32 BITS DIFFER FROM 'PVALVEST', THE RECORD OF ACTIONS TAKEN BY THIS ROUTINE, THE APPROPRIATE BITS IN 'CH5MASK' & 'CH6MASK', USED BY THE DAP JET-SELECTION LOGIC, ARE UPDATED, AS IS 'PVALVEST'. TO SPEED UP & SHORTEN THE ROUTINE, NO MORE THAN ONE CHANGE IS ACCEPTED PER ENTRY. THE HIGHEST-NUMBERED BIT IN CHANNEL 32 WHICH REQUIRES ACTION IS THE ONE PROCESSED.

THE CODING IN THE FAILURE MONITOR HAS BEEN WRITTEN SO AS TO HAVE ALMOST COMPLETE RESTART PROTECTION. FOR EXAMPLE, NO ASSUMPTION IS MADE WHEN SETTING A 'CH5MASK' BIT TO 1 THAT THE PREVIOUS STATE IS 0, ALTHOUGH IT OF COURSE SHOULD BE. ONE CASE WHICH MAY BE SEEN TO EVADE PROTECTION IS THE OCCURRENCE OF A RESTART AFTER UPDATING ONE OR BOTH DAP MASK-WORDS BUT BEFORE UPDATING 'PVALVEST', COUPLED WITH A CHANGE IN THE VALVE-BIT BACK TO ITS FORMER STATE. THE CONSEQUENCE OF THIS IS THAT THE NEXT ENTRY WOULD NOT SEE THE CHANGE INCOMPLETELY INCORPORATED BY THE LAST PASS (BECAUSE IT WENT AWAY AT JUST THE RIGHT TIME), BUT THE DAP MASK-WORDS WILL BE INCORRECT. THIS COMBINATION OF EVENTS SEEMS QUITE REMOTE, BUT NOT IMPOSSIBLE UNLESS THE CREW OPERATES THE SWITCHES AT HALF-SECOND INTERVALS OR LONGER. IN ANY EVENT, A DISAGREEMENT BETWEEN REALITY AND THE DAP MASKS WILL BE CURED IF THE MISINTERPRETED SWITCH IS REVERSED AND THEN RESTORED TO ITS CORRECT POSITION (SLOWLY).

CALLING SEQUENCE:

TCF RCSMONIT (IN INTERRUPT MODE, EVERY 480 MS.)

EXIT: TCF RCSMONEX (ALL PATHS EXIT VIA SUCH AN INSTRUCTION)
RCSMONEX EQUALS RESUME

ERASABLE INITIALIZATION REQUIRED:

VIA FRESH START: PVALVEST = +0 (ALL JETS ENABLED)
CH5MASK, CH6MASK = +0 (ALL JETS OK)

OUTPUT: CH5MASK & CH6MASK UPDATED (1'S WHERE JETS NOT TO BE USED, IN CHANNEL 5 & 6 FORMAT)
PVALTEST UPDATED (1.5 WHEN VALVE CLOSURES HAVE BEEN TRANSLATED INTO CH5MASK & CH6MASK; CHAN 32 FORMAT)
JOB TO DO 1/ACCS.

DEBRIS: A, L, Q AND DEBRIS OF NOVAC.

SUBROUTINE CALLED: NOVAC.

EBANK= CH5MASK

BANK 23
SETLOC RCSMONT
BANK


```
1
2          COUNT*  $$/T4RCS
3
4  RCSMONIT  EQUALS  RCSMON
5
6  RCSMON    CS      ZERO
7
8          EXTEND
9  RXOR      CHAN32  # PICK UP + INVERT INVERTED CHANNEL 32.
10 MASK      LOW8    # KEEP JET-FAIL BITS ONLY.
11 TS        Q
12
13          CS      PVALVEST  # - -
14 MASK      Q        # FORM PC + PC.
15 TS        L        # (P = PREVIOUS ISOLATION VALVE STATE,
16 CS        Q        # C = CURRENT VALVE STATE (CH 32)).
17 MASK      PVALVEST
18 ADS       L        # RESULT NZ INDICATES ACTION REQUIRED.
19
20          EXTEND
21 BZF        RCSMONEX  # QUIT IF NO ACTION REQUIRED.
22
23          EXTEND
24 MP         BIT7     # MOVE BITS 8 - 1 OF A TO 14 - 7 OF L.
25 XCH        L        # ZERO TO L IN THE PROCESS.
26
27 -3         INCR     L
28 DOUBLE
29 OVSF       # BOUND TO GET OVERFLOW IN THIS LOOP.
30 TCF        -3      # SINCE WE ASSURED INITIAL NZ IN A.
31
32          INDEX    L
33 CA         BIT8 -1  # SAVE THE RELEVANT BIT (8 - 1).
34 TS        Q
35
36          MASK     PVALVEST  # LOOK AT PREVIOUS VALVE STATE BIT.
37 CCS       A
38 TCF       VOPENED  # THE VALVE HAS JUST BEEN OPENED.
39
40          CS       CH5MASK   # THE VALVE HAS JUST BEEN CLOSED.
41 INDEX     L
42
43          MASK     5FAILTAB
44 ADS       CH5MASK   # SET INHIBIT BIT FOR CHANNEL 5 JET.
45
46          CS       CH6MASK
47 INDEX     L
48 MASK     6FAILTAB
49 ADS       CH6MASK   # SET INHIBIT BIT FOR CHANNEL 6 JET.
50
51          CA       Q
52 ADS       PVALVEST  # RECORD ACTION TAKEN.
53
54          TCF      1/ACCFIX  # SET UP 1/ACCJOB AND EXIT.
55
56
57
58
59
60
```

VOPENED	INDEX	L	# A VALVE HAS JUST BEEN OPENED.
	CS	5FAILTAB	
	MASK	CH5MASK	
	TS	CH5MASK	# REMOVE INHIBIT BIT FOR CHANNEL 5 JET.
	INDEX	L	
	CS	6FAILTAB	
	MASK	CH6MASK	
	TS	CH6MASK	# REMOVE INHIBIT BIT FOR CHANNEL 6 JET.
	CS	Q	
	MASK	PVALVEST	
	TS	PVALVEST	# RECORD ACTION TAKEN.
1/ACCFIX	CAF	PRI027	# SET UP 1/ACCS SO THAT THE SWITCH CURVES
	TC	NOVAC	# FOR TJETLAW CAN BE MODIFIED IF CH5MASK
	EBANK=	AOSQ	# HAS BEEN ALTERED.
	2CADR	1/ACCJOB	
	TCF	RCSMONEX	# EXIT.
5FAILTAB	EQUALS	-1	# CH 5 JET BIT CORRESPONDING TO CH 32 BIT:
	OCT	00040	# 8
	OCT	00020	# 7
	OCT	00100	# 6
	OCT	00200	# 5
	OCT	00010	# 4
	OCT	00001	# 3
	OCT	00004	# 2
	OCT	00002	# 1
6FAILTAB	EQUALS	-1	# CH 6 JET BIT CORRESPONDING TO CH 32 BIT:
	OCT	00010	# 8
	OCT	00020	# 7
	OCT	00004	# 6
	OCT	00200	# 5
	OCT	00001	# 4
	OCT	00002	# 3
	OCT	00040	# 2
	OCT	00100	# 1

BANK 22
SETLOC DOWNTLM
BANK

EBANK= DNTMBUFF

SPECIAL DOWNLINK OP CODES

#	OP CODE	ADDRESS (EXAMPLE)	SENDS...	BIT 15	BITS 14-12	BITS 11-0
#	-----	-----	-----	-----	-----	-----
#	1DNADR	TIME2	(2 AGC WDS)	0	0	ECADR
#	2DNADR	TEPHEM	(4 AGC WDS)	0	1	ECADR
#	3DNADR	VGBODY	(6 AGC WDS)	0	2	ECADR
#	4DNADR	STATE	(8 AGC WDS)	0	3	ECADR
#	5DNADR	UPBUFF	(10 AGC WDS)	0	4	ECADR
#	6DNADR	DSPTAB	(12 AGC WDS)	0	5	ECADR
#	DNCHAN	30	CHANNELS	0	7	CHANNEL ADDRESS
#	DNPTR	NEXTLIST	POINTS TO NEXT LIST	0	6	ADRES

- # DOWNLIST FORMAT DEFINITIONS AND RULES --
- # 1. END OF A LIST = -XDADR (X = 1 TO 6), -DNPTR, OR -DNCHAN.
 - # 2. SNAPSHOT SUBLIST = LIST WHICH STARTS WITH A -1DNADR.
 - # 3. SNAPSHOT SUBLIST CAN ONLY CONTAIN 1DNADRS.
 - # 4. TIME2 1DNADR MUST BE LOCATED IN THE CONTROL LIST OF A DOWNLIST.
 - # 5. ERASABLE DOWN TELEMETRY WORDS SHOULD BE GROUPED IN SEQUENTIAL LOCATIONS AS MUCH AS POSSIBLE TO SAVE STORAGE USED BY DOWNLINK LISTS.

	COUNT*	\$\$/DLIST	
ERASZERO	EQUALS	7	
UNKNOWN	EQUALS	ERASZERO	
SPARE	EQUALS	ERASZERO	# USE SPARE TO INDICATE AVAILABLE SPACE
LOWIDCOD	OCT	77340	# LOW ID CODE
NOMDNLST	EQUALS	LMCSTADL	# FRESH START AND POST P27 DOWNLIST
AGSLIST	EQUALS	LMAGSIDL	
UPDNLIST	EQUALS	LMAGSIDL	# UPDATE PROGRAM (P27) DOWNLIST

LM ORBITAL MANEUVERS LIST

#

----- CONTROL LIST -----

LMORBMDL	EQUALS	# SEND ID BY SPECIAL CODING
	DNPTR LMORBM01	# COLLECT SNAPSHOT
	6DNADR DNTMBUFF	# SEND SNAPSHOT
	1DNADR DELLT4	# DELLT4,+1
	3DNADR RTARG	# RTARG,+1...+5
	1DNADR ELEV	# ELEV,+1
	1DNADR TEVENT	# TEVENT,+1
	6DNADR REFSMMAT	# REFSMMAT +0...+11D
	1DNADR TCSI	# TCSI,+1
	3DNADR DELVEET1	# DELVEET1 +0...+5
	3DNADR VGTIG	# VGTIG +0...+5
	1DNADR DNLRLVELZ	# DNLRLVELZ,DNLRLALT
	1DNADR TPASS4	# TPASS4,+1
	DNPTR LMORBM02	# COMMON DATA
	1DNADR TIME2	# TIME2/1
	DNPTR LMORBM03	# COLLECT SNAPSHOT
	6DNADR DNTMBUFF	# SEND SNAPSHOT
	DNPTR LMORBM04	# COMMON DATA
	2DNADR POSTORKU	# POSTORKU,NEGORKU,POSTORKV,NEGORKV
	1DNADR SPARE	
	1DNADR TCDH	# TCDH,+1
	3DNADR DELVEET2	# DELVEET2 +0...+5
	1DNADR TTPI	# TTPI,+1
	3DNADR DELVEET3	# DELVEET3 +0...+5
	1DNADR DNRRANGE	# DNRRANGE,DNRRDOT
	2DNADR DNLRLVELX	# DNLRLVELX,DNLRLVELY,DNLRLVELZ,DNLRLALT
	1DNADR DIFFALT	# DIFFALT,+1
	1DNADR LEMMASS	# LEMMASS,CSMASS
	1DNADR IMODES30	# IMODES30,IMODES33
	1DNADR TIG	# TIG,+1
	DNPTR LMORBM05	# COMMON DATA
	DNPTR LMORBM06	# COMMON DATA
	1DNADR SPARE	# FORMERLY PIF
	-1DNADR TGO	# TGO,+1

----- SUB-LISTS -----

LMORBM01	-1DNADR R-OTHER +2	# R-OTHER +2,+3	SNAPSHOT
	1DNADR R-OTHER +4	# R-OTHER +4,+5	
	1DNADR V-OTHER	# V-OTHER,+1	
	1DNADR V-OTHER +2	# V-OTHER +2,+3	
	1DNADR V-OTHER +4	# V-OTHER +4,+5	
	1DNADR T-OTHER	# T-OTHER,+1	
	-1DNADR R-OTHER	# R-OTHER +0,+1	
LMORBM02	2DNADR REDOCTR	# REDOCTR,THETAD,+1,+2	COMMON DATA

1	# CHANNELR_21010			PAGE 100			1
2		1DNADR	RSBBQ	#	RSBBQ,+1		2
3		2DNADR	OMEGAP	#	OMEGAP,OMEGAQ,OMEGAR,GARBAGE		3
4		2DNADR	CDUXD	#	CDUXD,CDUYD,CDUZD,GARBAGE		4
5		2DNADR	CDUX	#	CDUX,CDUY,CDUZ,CDUT		5
6		6DNADR	STATE	#	STATE +0...+11D (FLAGWORDS)		6
7		-6DNADR	DSPTAB	#	DSPTAB TABLES		7
8							8
9	LMORBM03	-1DNADR	RN +2	#	RN +2,+3	SNAPSHOT	9
10		1DNADR	RN +4	#	RN +4,+5		10
11		1DNADR	VN	#	VN,+1		11
12		1DNADR	VN +2	#	VN +2,+3		12
13		1DNADR	VN +4	#	VN +4,+5		13
14		1DNADR	PIPTIME	#	PIPTIME,+1		14
15		-1DNADR	RN	#	RN,+1		15
16							16
17	LMORBM04	2DNADR	OMEGAPD	#	OMEGAPD,OMEGAQD,OMEGARD,GARBAGE		17
18		3DNADR	CADRFLSH	#	CADRFLSH,+1,+2,FAILREG,+1,+2		18
19		-1DNADR	RADMODES	#	RADMODES,DAPBOOLS	COMMON DATA	19
20							20
21	LMORBM05	2DNADR	OMEGAP	#	OMEGAP,OMEGAQ,OMEGAR,GARBAGE		21
22		2DNADR	CDUXD	#	CDUXD,CDUYD,CDUZD,GARBAGE		22
23		2DNADR	CDUX	#	CDUX,CDUY,CDUZ,CDUT		23
24		1DNADR	ALPHAQ	#	ALPHAQ,ALPHAR	COMMON DATA	24
25		1DNADR	POSTORKP	#	POSTORKP,NEGTORKP		25
26		DNCHAN	11	#	CHANNELS 11,12		26
27		DNCHAN	13	#	CHANNELS 13,14		27
28		DNCHAN	30	#	CHANNELS 30,31		28
29		-DNCHAN	32	#	CHANNELS 32,33		29
30							30
31	LMORBM06	1DNADR	PIPTIME1	#	PIPTIME,+1	COMMON DATA	31
32		-3DNADR	DELV	#	DELV +0...+5		32
33							33
34	#	-----					34
35							35
36							36
37							37
38							38
39							39
40							40
41							41
42							42
43							43
44							44
45							45
46							46
47							47
48							48
49							49
50							50
51							51
52							52
53							53
54							54
55							55
56							56
57							57
58							58
59							59
60							60

LM COAST AND ALIGNMENT DOWNLIST

#

----- CONTROL LIST -----

LMCSTADL	EQUALS		# SEND ID BY SPECIAL CODING
	DNPTR	LMCSTA01	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	1DNADR	AGSK	# AGSK,+1
	1DNADR	TALIGN	# TALIGN,+1
	2DNADR	POSTORKU	# POSTORKU,NEGORKU,POSTORKV,NEGORKV
	1DNADR	DNRRANGE	# DNRRANGE,DNRRDOT
	1DNADR	TEVENT	# TEVENT,+1
	6DNADR	REFSMAT	# REFSMAT +0...+11D
	1DNADR	AOTCODE	# AOTCODE,GARBAGE
	3DNADR	RLS	# RLS +0...+5
	2DNADR	DNLRVELX	# DNLRVELX,DNLRVELY,DNLRVELZ,DNLRALT
	DNPTR	LMCSTA06	# COMMON DATA
	DNPTR	LMCSTA02	# COMMON DATA
	1DNADR	TIME2	# TIME2/1
	DNPTR	LMCSTA03	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMCSTA04	# COMMON DATA
	DNPTR	LMCSTA07	# COMMON DATA
	2DNADR	DNLRVELX	# DNLRVELX,DNLRVELY,DNLRVELZ,DNLRALT
	2DNADR	CDUS	# CDUS,PIPAX,PIPAY,PIPAZ
	1DNADR	LASTYCMD	# LASTYCMD,LASTXCMD
	1DNADR	LEMMASS	# LEMMASS,CSMMASS
	1DNADR	IMODES30	# IMODES30,IMODES33
	1DNADR	TIG	# TIG,+1
	DNPTR	LMCSTA05	# COMMON DATA
	-6DNADR	DSPTAB	# DSPTAB +0...+11D TABLE

----- SUB-LISTS -----

LMCSTA01	EQUALS	LMORBM01	# COMMON DOWNLIST DATA
LMCSTA02	EQUALS	LMORBM02	# COMMON DOWNLIST DATA
LMCSTA03	EQUALS	LMORBM03	# COMMON DOWNLIST DATA
LMCSTA04	EQUALS	LMORBM04	# COMMON DOWNLIST DATA
LMCSTA05	EQUALS	LMORBM05	# COMMON DOWNLIST DATA
LMCSTA06	2DNADR	X789	# X789 +0...+3 COMMON DATA
	-1DNADR	LASTYCMD	# LASTYCMD,LASTXCMD
LMCSTA07	3DNADR	OGC	# OGC,+1,IGC,+1,MGC,+1 COMMON DATA
	1DNADR	BESTI	# BESTI,BESTJ
	3DNADR	STARSAV1	# STARSAV1 +0...+5
	-3DNADR	STARSAV2	# STARSAV2 +0...+5

LM RENDEZVOUS AND PRE-THRUST DOWNLIST

#

----- CONTROL LIST -----

LMRENDDL	EQUALS		# SEND ID BY SPECIAL CODING
	DNPTR	LMREND01	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMREND07	# COLLECT SNAPSHOT
	4DNADR	DNTMBUFF	# SEND SNAPSHOT
	1DNADR	DELLT4	# DELLT4,+1
	3DNADR	RTARG	# RTARG +0...+5
	3DNADR	DELVSLV	# DELVSLV +0...+5
	1DNADR	TCSI	# TCSI,+1
	3DNADR	DELVEET1	# DELVEET +0...+5
	1DNADR	SPARE	
	1DNADR	TPASS4	# TPASS4,+1
	DNPTR	LMREND06	# COMMON DATA
	DNPTR	LMREND02	# COMMON DATA
	1DNADR	TIME2	# TIME2/1
	DNPTR	LMREND03	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMREND04	# COMMON DATA
	2DNADR	POSTORKU	# POSTORKU,NEGORKU,POSTORKV,NEGORKV
	1DNADR	SPARE	
	1DNADR	TCDH	# TCDH,+1
	3DNADR	DELVEET2	# DELVEET2 +0...+5
	1DNADR	TTPI	# TTPI,+1
	3DNADR	DELVEET3	# DELVEET3 +0...+5
	1DNADR	ELEV	# ELEV,+1
	2DNADR	CDUS	# CDUS,PIPAX,PIPAY,PIPAZ
	1DNADR	LASTYCMD	# LASTYCMD,LASTXCMD
	1DNADR	LEMMASS	# LEMMASS,CSMASS
	1DNADR	IMODES30	# IMODES30,IMODES33
	1DNADR	TIG	# TIG,+1
	DNPTR	LMREND05	# COMMON DATA
	1DNADR	DELTAR	# DELTAR,+1
	1DNADR	CENTANG	# CENTANG,+1
	1DNADR	NN	# NN,+1
	1DNADR	DIFFALT	# DIFFALT,+1
	1DNADR	DELVTPF	# DELVTPF,+1
	-1DNADR	SPARE	

----- SUB-LISTS -----

LMREND01	EQUALS	LMORBM01	# COMMON DOWNLIST DATA
LMREND02	EQUALS	LMORBM02	# COMMON DOWNLIST DATA
LMREND03	EQUALS	LMORBM03	# COMMON DOWNLIST DATA



1					1			
2	LMREND04	EQUALS	LMORBM04	# COMMON DOWNLIST DATA	2			
3	LMREND05	EQUALS	LMORBM05	# COMMON DOWNLIST DATA	3			
4	LMREND06	EQUALS	LMCSTA06	# COMMON DOWNLIST DATA	4			
5					5			
6	LMREND07	-1DNADR	AIG	# AIG,AMG SNAPSHOT	6			
7		1DNADR	AOG	# AOG,TRKMKCNT	7			
8		1DNADR	TANGNB	# TANGNB,+1	8			
9		1DNADR	MKTIME	# MKTIME,+1	9			
10		-1DNADR	RANGRDOT	# DNRRANGE,DNRRDOT	10			
11					11			
12	#	-----			12			
13					13			
14					14			
15					15			
16					16			
17					17			
18					18			
19					19			
20					20			
21					21			
22					22			
23					23			
24					24			
25					25			
26					26			
27					27			
28					28			
29					29			
30					30			
31					31			
32					32			
33					33			
34					34			
35					35			
36					36			
37					37			
38					38			
39					39			
40					40			
41					41			
42					42			
43					43			
44					44			
45					45			
46					46			
47					47			
48					48			
49					49			
50					50			
51					51			
52					52			
53					53			
54					54			
55					55			
56					56			
57					57			
58					58			
59					59			
60					60			

LM DESCENT AND ASCENT DOWNLIST

----- CONTROL LIST -----

LMDSASDL	EQUALS		# SEND ID BY SPECIAL CODING
	DNPTR	LMDSAS07	# COLLECT SNAPSHOT
	DNPTR	LMDSAS08	# SEND SNAPSHOT
	1DNADR	TEVENT	# TEVENT,+1
	3DNADR	UNFC/2	# UNFC/2 +0...+5
	3DNADR	VGVECT	# VGVECT +0...+5
	1DNADR	TTF/8	# TTF/8,+1
	1DNADR	DELTAH	# DELTAH,+1
	3DNADR	RLS	# RLS +0...+5
	1DNADR	SPARE	
	DNPTR	LMDSAS09	# COMMON DATA
	DNPTR	LMDSAS02	# COMMON DATA
	1DNADR	TIME2	# TIME2/1
	DNPTR	LMDSAS03	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMDSAS04	# COMMON DATA
	2DNADR	POSTORKU	# POSTORKU,NEGORKU,POSTORKV,NEGORKV
	3DNADR	RGU	# RGU +0...+5
	3DNADR	VGU	# VGU +0...+5
	3DNADR	LAND	# LAND +0...+5
	1DNADR	AT	# AT,+1
	1DNADR	TLAND	# TLAND,+1
	1DNADR	FC	# FC,GARBAGE
	1DNADR	LASTYCMD	# LASTYCMD,LASTXCMD
	1DNADR	LEMMASS	# LEMMASS,CSMASS
	1DNADR	IMODES30	# IMODES30,IMODES33
	1DNADR	TIG	# TIG,+1
	DNPTR	LMDSAS05	# COMMON DATA
	DNPTR	LMDSAS06	# COMMON DATA
	1DNADR	PSEUDO55	# PSEUDO55,GARBAGE
	-1DNADR	TTOGO	# TTOGO,+1

----- SUB-LISTS -----

LMDSAS02	EQUALS	LMORBM02	# COMMON DOWNLIST DATA
LMDSAS03	EQUALS	LMORBM03	# COMMON DOWNLIST DATA
LMDSAS04	EQUALS	LMORBM04	# COMMON DOWNLIST DATA
LMDSAS05	EQUALS	LMORBM05	# COMMON DOWNLIST DATA
LMDSAS06	EQUALS	LMORBM06	# COMMON DOWNLIST DATA
LMDSAS07	-1DNADR	LRZCDUDL	# LRZCDUDL,GARBAGE
	1DNADR	VSELECT	# VSELECT,GARBAGE
	1DNADR	LRVTIMDL	# LRVTIMDL,+1

SNAPSHOT

1412THE

1

LM LUNAR SURFACE ALIGN DOWNLIST

----- CONTROL LIST -----

LMLSALDL	EQUALS		# SEND ID BY SPECIAL CODING
	DNPTR	LMLSAL01	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMLSAL07	# COLLECT SNAPSHOT
	4DNADR	DNTMBUFF	# SEND SHAPSHOT
	1DNADR	TALIGN	# TALIGN,+1
	6DNADR	REFSMMAT	# REFSMMAT +0...+11D
	6DNADR	YNBSAV	# YNBSAV +0...+5,SNBSAV +0...+5
	DNPTR	LMLSAL08	# COMMON DATA
	DNPTR	LMLSAL02	# COMMON DATA
	1DNADR	TIME2	# TIME2/1
	DNPTR	LMLSAL03	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SHAPSHOT
	DNPTR	LMLSAL04	# COMMON DATA
	DNPTR	LMLSAL09	# COMMON DATA
	3DNADR	GSAV	# GSAV +0...+5
	1DNADR	AGSK	# AGSK,+1
	1DNADR	LASTYCMD	# LASTYCMD, LASTXCMD
	1DNADR	LEMMASS	# LEMMASS,CSMMASS
	1DNADR	IMODES30	# IMODES30,IMODES33
	1DNADR	TIG	# TIG,+1
	DNPTR	LMLSAL05	# COMMON DATA
	DNPTR	LMLSAL06	# COMMON DATA
	1DNADR	SPARE	
	-1DNADR	SPARE	

----- SUB-LISTS -----

LMLSAL01	EQUALS	LMORBM01	# COMMON DOWNLIST DATA
LMLSAL02	EQUALS	LMORBM02	# COMMON DOWNLIST DATA
LMLSAL03	EQUALS	LMORBM03	# COMMON DOWNLIST DATA
LMLSAL04	EQUALS	LMORBM04	# COMMON DOWNLIST DATA
LMLSAL05	EQUALS	LMORBM05	# COMMON DOWNLIST DATA
LMLSAL06	EQUALS	LMORBM06	# COMMON DOWNLIST DATA
LMLSAL07	EQUALS	LMREND07	# COMMON DOWNLIST DATA
LMLSAL08	EQUALS	LMCSTA06	# COMMON DOWNLIST DATA
LMLSAL09	EQUALS	LMCSTA07	# COMMON DOWNLIST DATA

LM AGS INITIALIZATION AND UPDATE DOWNLIST

----- CONTROL LIST -----

LMAGSIDL	EQUALS		# SEND IO BY SPECIAL CODING
	3DNADR	AGSBUFF +0	# AGBUFF +0...+5
	1DNADR	AGSBUFF +12D	# AGBUFF +12D,GARBAGE
	3DNADR	AGSBUFF +1	# AGBUFF +1...+6
	1DNADR	AGSBUFF +13D	# AGBUFF +13D, GARBAGE
	3DNADR	AGSBUFF +6	# AGBUFF +6...+11
	1DNADR	AGSBUFF +12D	# AGBUFF +12D,GARBAGE
	3DNADR	AGSBUFF +7	# AGBUFF +7...+12D
	1DNADR	AGSBUFF +13D	# AGBUFF +13D,GARBAGE
	6DNADR	COMPNUMB	# COMPNUMB,UPOLDMOD,UPVERB,UPCOUNT,
			# UPBUFF +0...+7
	6DNADR	UPBUFF +8D	# UPBUFF +8D...+19D
	DNPTR	LMAGSI02	# COMMON DATA
	1DNADR	TIME2	# TIME2/1
	DNPTR	LMAGSI03	# COLLECT SNAPSHOT
	6DNADR	DNTMBUFF	# SEND SNAPSHOT
	DNPTR	LMAGSI04	# COMMON DATA
	2DNADR	POSTORKU	# POSTORKU,NEGTORKU,POSTORKV,NEGTORKV
	1DNADR	SPARE	
	1DNADR	SPARE	
	1DNADR	AGSK	# AGSK,+1
	6DNADR	UPBUFF	# UPBUFF +0...+11D
	4DNADR	UPBUFF +12D	# UPBUFF +12D...+19D
	1DNADR	LEMMASS	# LEMMASS,CSMMASS
	1DNADR	IMODES30	# IMODES30,IMODES33
	1DNADR	SPARE	
	DNPTR	LMAGSI05	# COMMON DATA
	-6DNADR	DSPTAB	# DSPTAB +0...+11D

----- SUB-LISTS -----

LMAGSI02	EQUALS	LMORBM02	# COMMON DOWNLIST DATA
LMAGSI03	EQUALS	LMORBM03	# COMMON DOWNLIST DATA
LMAGSI04	EQUALS	LMORBM04	# COMMON DOWNLIST DATA
LMAGSI05	EQUALS	LMORBM05	# COMMON DOWNLIST DATA

DNTABLE	GENADR	LMCSTADL	# LM COAST AND ALIGN DOWNLIST
	GENADR	LMAGSIDL	# LM AGS INITIALIZATION/UPDATE DOWNLIST
	GENADR	LMRENDDL	# LM RENDEZVOUS AND PRE-THRUST DOWNLIST
	GENADR	LMORBMDL	# LM ORBITAL MANEUVERS DOWNLIST
	GENADR	LMDSASDL	# LM DESCENT AND ASCENT DOWNLIST



DOWNLINK_LISTS



1412THE

1	GENADR LMLSALDL		# LM LUNAR SURFACE ALIGN DOWNLIST	1
2				2
3				3
4	#	-----		4
5				5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

PROGRAM NAME: AGS INITIALIZATION (R47)

#

WRITTEN BY: RHODE/KILROY/FOLLETT

#

MOD NO.: 0

DATE: 23 MARCH 1967

MOD BY: KILROY

#

MOD NO.: 1

DATE: 28 OCTOBER 1967

MOD BY: FOLLETT

#

FUNCT. DESC.: (1) TO PROVIDE THE AGS ABORT ELECTRONICS ASSEMBLY (AEA) WITH THE LEM AND CSM STATE VECTORS

(POSITION,VELOCITY,TIME) IN LEM IMU COORDINATES BY MEANS OF THE LGC DIGITAL DOWNLINK.

#

(2) TO ZERO THE ICDU, LGC, AND AEA GIMBAL ANGLE COUNTER SIMULTANEOUSLY IN ORDER TO ESTABLISH A

COMMON ZERO REFERENCE FOR THE MEASUREMENT OF GIMBAL (EULER) ANGLES WHICH DEFINE LEM ATTITUDE

(3) TO ESTABLISH THE GROUND ELAPSED TIME OF AEA CLOCK ZERO. (IF AN AEA CLOCK ZERO IS

REQUESTED DURING THIS PROGRAM

#

LOG SECTION: AGS INITIALIZATION

#

CALLING SEQ: PROGRAM IS ENTERED WHEN ASTRONAUT KEYS V47E ON DSKY.

R47 MAY BE CALLED AT ANY TIME EXCEPT WHEN ANOTHER EXTENDED VERB IS IN PROGRESS

#

SUBROUTINES

CALLED:

#

NORMAL EXIT: ENDEXT

#

ALARM/ABORT: ALARM -- BAD REFSMMAT -- CODE:220

OPERATOR ERROR IF V47 SELECTED DURING ANOTHER EXTENDED VERB.

#

ERASABLES

USED: SAMPTIME (2) TIME OF :ENTER: KEYSTROKE

AGSK (2) GROUND ELAPSED TIME OF THE AEA CLOCK :ZERO:

AGSBUFF (14D) CONTAINS AGS INITIALIZATION DATA (SEE :OUTPUT: BELOW)

AGSWORD (1) PREVIOUS DOWNLIST SAVED HERE

EBANK= AGSBUFF

BANK 40

SETLOC R47

BANK

COUNT* \$\$/R47

AGSINIT

CAF REFSMBIT

MASK FLAGWRD3

CCS A

CHECK REFSMFLG.

	TC	REDSPTM	# REFSMMAT IS OK
	TC	ALARM	# REFSMMAT IS BAD
	OCT	220	
	TC	ENDEXT	
NEWAGS	EXTEND		
	DCA	SAMPTIME	# TIME OF THE :ENTER: KEYSTROKE
	DXCH	AGSK	# BECOMES NEW AEA CLOCK :ZERO:
REDSPTM	EXTEND		
	DCA	AGSK	
AGSDISPK	DXCH	DSPTMX	
	CAF	V06N16	
	TC	BANKCALL	# R1 = 00XXX. HRS., R2 = 000XX MIN.,
	CADR	GOMARKF	# R3 = 0XX.XX SEC.
	TC	ENDEXT	# TERMINATE RETURN
	TC	AGSVCALC	# PROCEED RETURN
	CS	BIT6	# IS ENTER VIA A V32
	AD	MPAC	
	EXTEND		
	BZF	NEWAGS	# YES, USE KEYSTROKE TIME FOR NEW AGSK
	EXTEND		# NO, NEW AGSK LOADED VIA V25
	DCA	DSPTMX	# LOADED INTO DSPTMX BY KEYING
	TC	REDSPTM -1	# V25E FOLLOWED BY HRS.,MINS.,SECS.
			# DISPLAY THE NEW K
AGSVCALC	TC	INTPRET	
	SET		
	SET	NODOFLAG	# DON'T ALLOW V37
		EXIT	
		XDSPFLAG	
	CAF	V06N16	
	TC	BANKCALL	
	CADR	EXDSPRET	
	TC	INTPRET	# EXTRAPOLATE LEM AND CSM STATE VECTORS
	RTB		# TO THE PRESENT TIME
		LOADTIME	# LOAD MPAC WITH TIME2,TIME1
	STCALL	TDEC1	# CALCULATE LEM STATE VECTOR
		LEMPREC	
	CALL		# CALL ROUTINE TO CONVERT TO SM COORDS AND
		SCALEVEC	# PROVIDE PROPER SCALING
	STODL	AGSBUFF	# (LEMPREC AND CSMPREC LEAVE TDEC1 IN TAT)
		TAT	# TAT = TIME TO WHICH RATT1 AND VATT1 ARE
	STCALL	TDEC1	# COMPUTED (CSEC SINCE CLOCK START B-28).
		CSMPREC	# CALCULATE CSM STATE VECTOR FOR SAME TIME
	CALL		
		SCALEVEC	

	STODL	AGSBUFF +6 TAT	
	DSU	DDV	# CALCULATE AND STORE THE TIME
		AGSK	
		TSCALE	
	STORE EXIT	AGSBUFF +12D	
	CAF TS	LAGSLIST DNLSTCOD	
	CAF TC CADR	20SEC BANKCALL DELAYJOB	# DELAY FOR 20 SEC WHILE THE AGS # DOWNLIST IS TRANSMITTED
	CA TS	AGSWORD DNLSTCOD	# RETURN TO THE OLD DOWNLIST
	CAF MASK CCS	IMUSEBIT FLAGWRDO A	# CHECK IMUSE FLAG.
CKSTALL	TC CCS TCF	AGSEND IMUCADR +3	# IMU IS BEING USED -- DO NOT ZERO # CHECK FOR IMU USAGE WHICH AVOIDS THE # IMUSE BIT: I.E., IMU COMPENSATION.
	TCF TCF	+6 +1	# FREE. GO AHEAD WITH THE IMU ZERO.
+3	CAF	TEN	# WAIT .1 SEC AND TRY AGAIN.
	TC CADR TCF	BANKCALL DELAYJOB CKSTALL	
+6	TC CADR	BANKCALL IMUZERO	# IMU IS NOT IN USE # SET IMU ZERO DISCRETE FOR 320 MSECS.
	TC CADR TC	BANKCALL IMUSTALL AGSEND	# WAIT 3 SEC FOR COUNTERS TO INCREMENT
AGSEND	TC ADRES	DOWNFLAG NODOFLAG	# ALLOW V37
	CAF TC CADR	V50N16 BANKCALL GOMARK3	
	TCF TCF TC	ENDEXT ENDEXT ENDEXT	
SCALEVEC	VLOAD	MXV VATT1	
	VXSC	REFSMMAT VSL2 VSCALE	

```
# THIS SECTION ROUNDS THE VECTOR, AND
# CORRECTS FOR THE FACT THAT THE AGS
# IS A 2 S COMPLEMENT MACHINE WHILE THE
# LGC IS A 1 S COMPLEMENT MACHINE.
```

```
# AGAIN THIS SECTION ROUNDS. TWO VECTORS
# ARE ADDED TO DEFEAT ALSIGNAG IN THE
# CASE OF A HIGH-ORDER ZERO COUPLED WITH
# A LOW ORDER NEGATIVE PART.
```

CSEC TO SEC SCALE FACTOR

```
# METERS TO FEET SCALE FACTOR
# METERS/CS TO FEET/SEC SCALE FACTOR
```



AGS_INITIALIZATION

PAGE 210

1412THE

1				1
2	20CT	0000037777		2
3				3
4	SBANK=	LOWSUPER	# FOR SUBSEQUENT LOW 2CADRS.	4
5				5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

```
1
2      BANK      10
3      SETLOC    FRANDRES
4      BANK
5
6      EBANK=     LST1
7
8      SLAP1      COUNT*  $$/START      # FRESH AND RESTART
9              INHINT      # FRESH START.  COMES HERE FROM PINBALL.
10             TC          STARTSUB      # SUBROUTINE DOES MOST OF THE WORK
11
12      STARTSW    TCF       SKIPSIM      # PATCH....TCF STARTSIM FOR SIMULATION
13      STARTSIM   CAF       BIT14
14             TC       FINDVAC
15      SIM2CADR   OCT       77777      # PATCH 2CADR (AND EBANK DESIGNATION) OF
16             OCT       77777      # SIMULATION START ADDRESS.
17
18      SKIPSIM     CA        DSPTAB +11D  # TURN OFF ALL DSPTAB +11D LAMPS
19             MASK      BITS4&6      # EXCEPT THE GIMBAL LOCK & NO ATT ONLY ON
20             AD        BIT15      # REQUESTED FRESH START.
21             TS        DSPTAB +11D
22
23             CA        BIT12      # INITIALIZE DOWNLINK EARASABLE MEMORY
24             TS        DUMPCNT      # DUMP FOR ONE PASS
25
26             CA        ZERO
27             TS        ERCOUNT
28             TS        FAILREG
29             TS        FAILREG +1
30             TS        FAILREG +2
31             TS        REDOCTR
32
33             CS        PRI012
34             TS        DSRUPTSW
35
36      DOFSTART    CAF       BIT14      # INSURE ENGINE IS OFF.
37             EXTEND
38             WRITE    DSALMOUT
39             CS        ZERO
40             TS        THRUST
41
42      DOFSTRT1    CAF       FOUR
43             TS        RCSFLAGS      # INITIALIZE ATTITUDE ERROR DISPLAYS.
44             CA        PRI030
45             TS        RESTREG      # SUPER BANK PRIORITY FOR DISPLAYS.
46
47             CA        ZERO
48             TS        ABDELV      # DAP INITIALIZATION
49             TS        NVSAVE
50             TS        EBANKTEM
51
52
53
54
55
56
57
58
59
60
```

1				
2	TS	CH5MASK		
3	TS	CH6MASK		
4	TS	PVALVEST	# FOR RCS FAILURE MONITOR	
5	TS	ERESTORE	# ***** MUST NOT BE REMOVED FROM DOFSTART	
6	TS	SMODE	# ***** MUST NOT BE REMOVED FROM DOFSTART	
7	TS	DNLSTCOD	# SELECT P00 DOWNLIST	
8	TS	AGSWORD	# ALLOW AGS INITIALIZATION	
9	TS	UPSVFLAG	# ZERO UPDATE STATE VECTOR REQUEST FLAGWRD	
10	EXTEND			
11	WRITE	CHAN5	# TURN OFF RCS JETS.	
12	EXTEND			
13	WRITE	CHAN6	# TURN OFF RCS JETS.	
14	EXTEND			
15	WRITE	CHAN12		
16	EXTEND			
17	WRITE	CHAN13		
18	EXTEND			
19	WRITE	CHAN14		
20	CS	DSPTAB +11D		
21	MASK	BITS4&6		
22	CCS	A		
23	TC	+4		
24	CA	BITS4&6		
25	EXTEND		# THE IMU WAS IN COARSE ALIGN IN GIMBAL	
26	WOR	CHAN12	# LOCK, SO PUT IT BACK INTO COARSE ALIGN.	
27	TC	MR.KLEAN		
28				
29	CS	ZERO		
30	TS	MODREG		
31				
32	CAF	IM30INIF	# FRESH START IMU INITIALIZATION	
33	TS	IMODES30		
34				
35	CAF	MAXDB		
36	TS	DB		
37	CAF	FOUR		
38	TS	RATEINDX	# INITIALZE KALCMANU RATE	
39	CA	BOOLSTRT		
40	TS	DAPBOOLS		
41	CAF	EBANK6		
42	TS	EBANK		
43	EBANK=	HIASCENT		
44				
45	CA	STIKSTRT		
46	TS	STIKSENS		
47	CA	RATESTRT		
48	TS	-RATEDB		
49	CAF	FULLAPS	# INITIALIZE MAXIMUM ASCENT MASS FOR USE	
50	TS	HIASCENT	# BY 1/ACCS UNTIL THE PAD LOAD IS DONE.	
51	CA	77001OCT	# LOAD DAP FILTER GAINS PAD LOAD.	
52				
53				
54				
55				
56				
57				
58				
59				
60				

```
1
2      TS      DKTRAP      #      TO BEST PRESENT ESTIMATE OF GOODIES
3      TS      LMTRAP      # .14 DEG
4
5      CA      60DEC
6      TS      DKKAOSN
7      TS      LMKAOSN      # 6 SEC GAIN FOR ALPHA
8
9      CA      ZERO
10     TS      LMOMEGAN      # UNITY GAIN
11     CA      TEN
12
13     TS      DKOMEGAN      # 1 SEC GAIN FOR OMEGA
14     CAF     BIT8          # SET DOCKED DB TO 1.4 DEG.  MAY OVERWRITE
15     TS      DKDB          #      WITH PAD LOAD.
16
17     CAF     IM33INIT
18     AD      BIT6          # KEEP BOTH DAP AND ERROR-NEEDLES DISPLAY
19     TS      IMODES33      #      OFF UNTIL ICDU ZERO IS FINISHED.
20
21
22     EXTEND
23     DCA      SWINIT      # INITIALIZE SWITCHES ONLY ON FRESH START.
24
25     DXCH     STATE
26     CA      SWINIT +2
27     TS      STATE +2
28
29     CA      REFSMBIT      # DO NOT ALTER REFSMFLG ON FRESH START.
30     MASK     STATE +3
31     AD      SWINIT +3
32
33     TS      STATE +3
34
35     EXTEND
36     DCA      SWINIT +4
37
38     DXCH     STATE +4
39
40     EXTEND
41     DCA      SWINIT +6
42
43     DXCH     STATE +6
44     CA      SURFFBIT      # DO NOT ALTER SURFFLAG ON FRESH START.
45     AD      CMOONBIT      #      CMOONFLG
46
47     AD      LMOONBIT      #      LMOONFLG
48     MASK     STATE +8D
49     AD      SWINIT +8D
50
51     TS      STATE +8D
52     CA      SWINIT +9D
53     TS      STATE +9D
54
55     CA      APSFLBIT      # DO NOT ALTER APSFLAG ON FRESH START.
56     MASK     STATE +10D
57     AD      SWINIT +10D
58
59     TS      STATE +10D
60     CAF     SWINIT +11D
61     TS      STATE +11D
62
63
64     ENDRSTRT      TC      POSTJUMP      # NOW IN ANOTHER BANK.
65     CADR      DUMMYJOB +2      # PICKS UP AT RELINT.  (DON'T ZERO NEWJOB)
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

```
MR.KLEAN      INHINT
```


COMES HERE FROM LOCATION 4000, GOJAM, RESTART ANY PROGRAMS WHICH MAY HAVE BEEN RUNNING AT THE TIME.

GOPROG EBANK= LST1
 INCR REDOCTR # ADVANCE RESTART COUNTER.

LXCH Q
EXTEND
ROR SUPERBNK

DXCH RSBBQ
CA DSPTAB +11D
MASK BIT4

EXTEND
BZF +4
AD BIT6 # SET ERROR COUNTER ENABLE

 EXTEND
 WOR CHAN12 # ISS WAS IN COARSE ALIGN SO GO BACK TO
BUTTONS TC LIGHTSET

ERASCHK TEMPORARILY STORES THE CONTENST OF TWO ERASABLE LOCATIONS, X
AND X+1 INTO SKEEP5 AND SKEEP6. IT ALSO STORES X INTO SKEEP7 AND
ERESTORE. IF ERASCHK IS INTERRUPTED BY A RESTART, C(ERESTORE) SHOULD
EQUAL C(SKEEP7), AND SHOULD BE A + NUMBER LESS THAN 2000 OCT. OTHERWISE
C(ERESTORE) SHOULD EQUAL +0.

CAF HI5
MASK ERESTORE

EXTEND
BZF +2 # IF ERESTORE NOT = +0 OR +N LESS THAN 2K,
TCF NONAVKEY +3 # DO FRESH START -- E MEMORY MIGHT BE BAD

CS ERESTORE

EXTEND
BZF DORSTART # = +0 CONTINUE WITH RESTART.

AD SKEEP7

EXTEND
BZF +2 # = SKEEP7, RESTORE E MEMORY.

TCF NONAVKEY +3 # DO FRESH START -- E MEMORY MIGHT BE BAD

CA SKEEP4

TS EBANK # EBANK OF E MEMORY THAT WAS UNDER TEST.

 # (NOT DXCH SINCE THIS MIGHT HAPPEN AGAIN)

EXTEND

DCA SKEEP5

INDEX SKEEP7

DXCH 0000 # E MEMORY RESTORED

CA ZERO

TS ERESTORE

DORSTART TC STARTSUB # DO INITIALIZATION AFTER ERASE RESTORE.

SETINFL CS INTFLBIT

MASK FLGWRD10

TS FLGWRD10

1				
2		CA	9,6,4	# LEAVE PROG ALARM, GIMBAL LOCK, NO ATT
3		MASK	DSPTAB +11D	# LAMPS INTACT ON HARDWARE RESTART
4		AD	BIT15	
5		XCH	DSPTAB +11D	
6		CAF	IFAILINH	# LEAVE IMU FAILURE INHIBITS INTACT ON
7		MASK	IMODES30	# HARDWARE RESTART, RESET ALL FAILURE
8		AD	IM30INIR	# CODES.
9		TS	IMODES30	
10				
11		CA	AGSWORD	# BE SURE OF CORRECT DOWNLIST
12		TS	DNLSTCOD	
13				
14		CA	BIT4	# TURN ON THROTTLE COUNTER
15		EXTEND		
16		WOR	CHAN14	# TURN ON THRUST DRIVE
17		CS	FLAGWRD5	
18		MASK	ENGONBIT	
19		CCS	A	
20		TCF	+5	
21		CAF	BIT13	
22		EXTEND		
23		WOR	DSALMOUT	# TURN ENGINE ON
24		TCF	GOPROG3	
25	+5	CAF	BIT14	
26		EXTEND		
27		WOR	DSALMOUT	# TURN ENGINE OFF
28		TCF	GOPROG3	
29				
30	ENEMA	INHINT		
31		TC	STARTSB1	
32		TCF	GOPROG2A	
33	GOPROG2	TC	STARTSB2	
34	GOPROG2A	TC	LIGHTSET	
35		CS	RSFLGBTS	# CLEAR BITS 7 AND 14.
36		MASK	FLGWRD10	
37		TS	FLGWRD10	
38				
39	GOPROG3	CAF	NUMGRPS	# VERIFY PHASE TABLE AGREEMENTS
40	PCLOOP	TS	MPAC +5	
41		DOUBLE		
42		EXTEND		
43		INDEX	A	
44		DCA	-PHASE1	# COMPLEMENT INTO A, DIRECT INTO L.
45		EXTEND		
46		RXOR	LCHAN	# RESULT MUST BE -0 FOR AGREEMENT.
47		CCS	A	
48		TCF	PTBAD	# RESTART FAILURE.
49		TCF	PTBAD	
50		TCF	PTBAD	
51				
52				
53				
54				
55				
56				
57				
58				
59				
60				

1					1
2		CCS	MPAC +5	# PROCESS ALL RESTART GROUPS.	2
3		TCF	PCLOOP		3
4					4
5		TS	MPAC +6	# SET TO +0.	5
6		TC	MMDSPY	# DISPLAY MAJOR MODE	6
7					7
8		INHINT		# RELINT DONE IN MMDSPY	8
9					9
10		CS	DIDFLBIT	# CLEAR DIDFLAG IN ORDER TO FORCE R10 TO	10
11		MASK	FLAGWRD1	# RE-INITIALIZE ITSELF IF IT HAD BEEN	11
12		TS	FLAGWRD1	# OPERATION AT THE TIME OF THE RESTART.	12
13					13
14		CS	RODFLBIT	# CLEAR RODFLAG. IF P66 IS IN OPERATION	14
15		MASK	FLAGWRD1	# IT WILL RE-INITIALIZE ITSELF AND	15
16		TS	FLAGWRD1	# CONTINUE.	16
17					17
18		CS	P21FLBIT	# CLEAR P21 FLAG SO THAT P21 WILL COMPUTE	18
19		MASK	FLAGWRD0	# NEW BASE STATE VECTORS.	19
20		TS	FLAGWRD0		20
21					21
22		CAF	NUMGRPS	# SEE IF ANY GROUPS RUNNING.	22
23	NXTRST	TS	MPAC +5		23
24		DOUBLE			24
25		INDEX	A		25
26		CCS	PHASE1		26
27		TCF	PACTIVE	# PNZ -- GROUP ACTIVE.	27
28		TCF	PINACT	# +0 -- GROUP NOT RUNNING.	28
29					29
30	PACTIVE	TS	MPAC		30
31		INCR	MPAC	# ABS OF PHASE.	31
32		INCR	MPAC +6	# INDICATE GROUP DEMANDS PRESENT.	32
33		CA	RACTCADR		33
34		TC	SWCALL	# MUST RETURN TO SWRETURN.	34
35					35
36	PINACT	CCS	MPAC +5	# PROCESS ALL RESTART GROUPS.	36
37		TCF	NXTRST		37
38					38
39		CCS	MPAC +6	# NO, CHECK PHASE ACTIVITY FLAG	39
40		TCF	ENDRSTRT	# PHASE ACTIVE	40
41		CAF	BIT15	# IS MODE -0	41
42		MASK	MODREG		42
43		EXTEND			43
44		BZF	GOTOPOOH	# NO	44
45		TCF	ENDRSTRT	# YES	45
46	PTBAD	TC	ALARM	# SET ALARM TO SHOW PHASE TABLE FAILURE.	46
47		OCT	1107		47
48					48
49		TCF	DOFSTRT1		49
50	*****	*****	*****		50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

DO NOT USE GOPROG2 OR ENEMA WITHOUT CONSULTING POOH PEOPLE.

OCT10000	=	BIT13	
OCT30000	=	PRI030	
OCT7777	OCT	7777	
STIKSTRT	DEC	0.825268	# 20 D/S MAXIMUM COMPANDED RATE
RATESTRT	DEC	-218	
RACTCADR	CADR	RESTARTS	

BOOLSTRT	OCT	21312	
77001OCT	OCT	77001	# .14 DEG SCALED AT 4.5 DEG
60DEC	DEC	60	

RSFLGBTS	OCT	20100	
MAXDB	OCTAL	03434	# 5 DEG ATTITUDE DEADBAND, SCALED AT 45.

LIGHTSET	CAF	BIT5	# CHECK FOR MARK REJECT AND ERROR RESET
	EXTEND		
	RAND	NAVKEYIN	

	EXTEND		
	BZF	NONAVKEY	# NO MARK REJECT
	EXTEND		

	READ	MNKEYIN	# CHECK IF KEYS 2M AND 5M ON
	AD	-ELR	# MAIN DSKY KEYCODE (BITS 1-5)
	EXTEND		

	BZF	+2	
--	-----	----	--

NONAVKEY	TC	Q	
----------	----	---	--

	TC	STARTSUB	
	TCF	DOFSTART	
+3	TC	STARTSUB	
	TCF	DOFSTRT1	# DO FRESH START BUT DON'T TOUCH ENGINE

1412THE

INITIALIZATION COMMON TO BOTH FRESH START AND RESTART.

STARTSUB	EBANK= CAF TS	AOSQ LDNPHAS1 DNTMGOTO	# SET POINTER SO NEXT 20MS DOWNRUPT WILL # CAUSE THE CURRENT DOWNLIST TO BE # INTERRUPTED AND START SENDING FROM THE # BEGINNING OF THE CURRENT DOWNLIST.
	CAF	BIT6	
	EXTEND		
	RAND	CHAN33	
	AD	RMODINIT	
	TS	RADMODES	
STARTSB1	CAF	POSMAX	
	TS	TIME3	
	AD	MINUS2	
	TS	TIME4	
	AD	NEGONE	
	TS	TIME5	
	CAF	EBANK6	
	TS	EBANK	
	CS	BIT13	# CAUSE DAPIDLER TO CALL 1/ACCS
	MASK	RCSFLAGS	
	TS	RCSFLAGS	# ZERO BIT 13
	CAF	POSMAX	# DISABLE TIME6 CLOCK. JUST IN CASE A T6
	TS	T6NEXT	# RUPT IS ALREADY IN THE PRIORITY CHAIN,
	EXTEND		# ENSURE THAT ITS INPUTS WILL RENDER IT
	WAND	CHAN13	# INEFFECTUAL.
	CAF	ZERO	
	TS	NXT6ADR	
	TS	NEXTP	
	CS	ACCSOKAY	
	MASK	DAPBOOLS	
	TS	DAPBOOLS	
	EXTEND		# SET T5RUPT FOR DAPIDLER PROGRAM.
	DCA	IDLEADR	
	DXCH	T5ADR	
STARTSB2	CAF	OCT30001	# DURING SOFTWARE RESTART, DO NOT DISTURB
	EXTEND		# ENGINE ON, OFF AND ISS WARNING.
	WAND	DSALMOUT	
	CS	READRBIT	# CLEAR READRFLG FOR R29
	MASK	FLAGWRD3	
	TS	FLAGWRD3	

CS	FLAGWRD3	# DURING SOFTWARE RESTART, CLEAR TURNON,
MASK	NR29FBIT	# REPOSITION, CDU ZERO AND REMODE BITS
EXTEND		# IN RADMODES, SINCE TASKS ASSOCIATED
BZF	+2	# WITH THESE BITS HAVE BEEN KILLED
CAF	BIT10	# ALSO IF R29 HAD BEEN REQUESTED.
AD	OCT32001	# (NOR29FLG = 0) CLEAR BIT 10 RADMODES
COM		# TO MAKE R29 FORGET IT HAD STARTED
MASK	RADMODES	# DESIGNATING
TS	RADMODES	
CAF	OCT27470	# DURING SOFTWARE RESTART, DO NOT DISTURB
EXTEND		# IMU FLAGS. (COARSE ALIGN ENABLE, ZERO
WAND	CHAN12	# IMU CDUS, ENABLE IMU COUNTER) AND GIMBAL
		# TRIM DRIVES. LEAVE RR LOCKON ENABLE
		# ALONE.
CS	NORRMBIT	# ENABLE R25.
MASK	FLAGWRD5	
TS	FLAGWRD5	
CS	R77FLBIT	# CLEAR R77FLAG
MASK	FLAGWRD5	
TS	FLAGWRD5	
CAF	OCT74160	# DURING SOFTWARE RESTART, DO NOT DISTURB
EXTEND		# TELEMETRY FLAGS, RESET TRAP FLAGS, AND
WAND	CHAN13	# ENABLE T6RUPT FLAG.
CAF	BIT12	# REENABLE RUPT10 (RUPT QUICKLY
EXTEND		# RESUMES EXCEPT DURING P64)
WOR	CHAN13	
CAF	BIT6	# DURING SOFTWARE RESTART, DO NOT DISTURB
EXTEND		# GYRO ENABLE FLAG.
WAND	CHAN14	
EBANK=	LST1	
CAF	STARTEB	
TS	EBANK	# SET FOR E3
CAF	NEG1/2	# INITIALIZE WAITLIST DELTA-TS.
TS	LST1 +7	
TS	LST1 +6	
TS	LST1 +5	
TS	LST1 +4	
TS	LST1 +3	
TS	LST1 +2	
TS	LST1 +1	
TS	LST1	
CS	ENDTASK	
TS	LST2	

1	# FRESH START AND RESTART			1
2		TS	LST2 +2	2
3		TS	LST2 +4	3
4		TS	LST2 +6	4
5		TS	LST2 +8D	5
6		TS	LST2 +10D	6
7		TS	LST2 +12D	7
8		TS	LST2 +14D	8
9		TS	LST2 +16D	9
10		CS	ENDTASK +1	10
11		TS	LST2 +1	11
12		TS	LST2 +3	12
13		TS	LST2 +5	13
14		TS	LST2 +7	14
15		TS	LST2 +9D	15
16		TS	LST2 +11D	16
17		TS	LST2 +13D	17
18		TS	LST2 +15D	18
19		TS	LST2 +17D	19
20				20
21		CS	ZERO	21
22		TS	PRIORITY	22
23		TS	PRIORITY +12D	23
24		TS	PRIORITY +24D	24
25		TS	PRIORITY +36D	25
26		TS	PRIORITY +48D	26
27		TS	PRIORITY +60D	27
28		TS	PRIORITY +72D	28
29		TS	PRIORITY +84D	29
30				30
31		TS	DSRUPTSW	31
32		TS	NEWJOB	32
33				33
34		CAF	VAC1ADRC	34
35		TS	VAC1USE	35
36		AD	LTHVACA	36
37		TS	VAC2USE	37
38		AD	LTHVACA	38
39		TS	VAC3USE	39
40		AD	LTHVACA	40
41		TS	VAC4USE	41
42		AD	LTHVACA	42
43		TS	VAC5USE	43
44				44
45		CAF	TEN	45
46	DSPOFF	TS	MPAC	46
47		CS	BIT12	47
48		INDEX	MPAC	48
49		TS	DSPTAB	49
50		CCS	MPAC	50
51		TCF	DSPOFF	51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1			
2	TS	DELAYLOC	
3	TS	DELAYLOC +1	
4	TS	DELAYLOC +2	
5	TS	R1SAVE	
6	TS	INLINK	
7	TS	DSPCNT	
8	TS	CADRSTOR	
9	TS	REQRET	
10	TS	CLPASS	
11	TS	DSPLOCK	
12	TS	MONSAVE	# KILL MONITOR
13	TS	MONSAVE1	
14	TS	VERBREG	
15	TS	NOUNREG	
16	TS	DSPLIST	
17	TS	MARKSTAT	
18	TS	EXTVBACT	# MAKE EXTENDED VERBS AVAILABLE
19	TS	IMUCADR	
20	TS	OPTCADR	
21	TS	RADCADR	
22	TS	ATTCADR	
23	TS	LGYRO	
24	TS	FLAGWRD4	# KILL INTERFACE DISPLAYS
25	CAF	NOUTCON	
26	TS	NOUT	
27			
28	CS	ONE	
29	TS	SAMPLIM	
30	CAF	BIT6	
31	MASK	IMODES33	# LEAVE BIT 6 UNCHANGED
32	AD	IM33INIT	# NO PIP OR TM FAILS. BIT6=0 IN THIS WORD.
33	TS	IMODES33	
34			
35	CAF	LESCHK	# SELF CHECK GO-TO REGISTER.
36	TS	SELFRET	
37			
38	CS	VD1	
39	TS	DSPCOUNT	
40			
41	TC	Q	
42	EBANK=	AOSQ	
43	IDLEADR	2CADR	DAPIDLER
44			
45	IFAILINH	OCT	435
46	LDNPHAS1	GENADR	DNPHASE1
47	LESCHK	GENADR	SELFCHK
48	VAC1ADRC	ADRES	VAC1USE
49	OCT32001	OCT	32001
50	LTHVACA	DEC	44
51			
52			
53			
54			
55			
56			
57			
58			
59			
60			

1					1
2	OCT27470	OCT	27470		2
3	OCT74160	OCT	74160		3
4	OCT30001	OCT	30001		4
5	STARTEB	EQUALS	EBANK3		5
6	NUMGRPS	EQUALS	FIVE		6
7	-ELR	OCT	-22	# -ERROR LIGHT RESET KEY CODE.	7
8	IM30INIF	OCT	37411	# INHIBITS IMU FAIL FOR 5 SEC AND PIP ISSW	8
9	IM30INIR	OCT	37000		9
10	IM33INIT	=	PRI016	# NO PIP OR TM FAIL SIGNALS.	10
11	9,6,4	OCT	450		11
12	RMODINIT	OCT	00102		12
13	SWINIT	OCT	0		13
14		OCT	0		14
15		OCT	0		15
16		OCT	02000	# BIT 11 = NOR29FLG	16
17		OCT	0		17
18		OCT	0		18
19		OCT	0		19
20		OCT	00100		20
21					21
22		OCT	0		22
23		OCT	0		23
24		OCT	0		24
25		OCT	40000	# BIT 15 = LRBYPASS.	25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

FRESH_START_AND_RESTART

# PROGRAM NAME	GOTOPOOH	ASSEMBLY SUNDANCE
# LOG SECTION	FRESH START AND RESTART	

FUNCTIONAL DESCRIPTION

FLASH V 37 ON DSKY MM CHANGE REQUEST

INPUT/OUTPUT INFORMATION

#	A. CALLING SEQUENCE	TC GOTOPOOH
#	B. ERASABLE INITIALIZATION	NONE
#	C. OUTPUT FLASH V 37 ON DSKY	
#	D. DEBRIS	L

PROGRAM ANALYSIS

#	A. SUBROUTINES CALLED	PRIODSPR, LINUS
#	B. NORMAL EXIT	TCF ENDOFJOB
#	C. ALARM AND ABORT EXITS	NONE

BLOCK	03
SETLOC	FFTAG5
BANK	

GOTOPOOH	COUNT*	\$\$/P00	
	CAF	OCT33	# 4.33 SPOT FOR GOP00FIX
	TS	L	
	COM		
	DXCH	-PHASE4	

	TC	POSTJUMP
	CADR	GOP00FIX
OCT24	MM	20
OCT31	MM	25

BANK	20
SETLOC	VERB37
BANK	

GOP00FIX	COUNT*	\$\$/P00	# VERB 37 AND P00 IN BANK 4.
	TC	DOWNFLAG	# ALLOW X-AXIS OVERRIDE
	ADRES	XOVINFLG	
	TC	DOWNFLAG	# INSURE THAT ULLAGE IS OFF
	ADRES	ULLAGFLG	



FRESH_START_AND_RESTART

PAGE 225

1412THE

1				1
2		TC	CLEARMRK +2	2
3		CAF	V37N99	3
4		TC	BANKCALL	4
5		CADR	GOFLASH	5
6		TCF	-3	6
7		TCF	-4	7
8		TCF	-5	8
9				9
10	V37N99	VN	3799	10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

```
1 # PROGRAM NAME          V37          ASSEMBLY SUNDANCE
2 #
3 # LOG SECTION            FRESH START AND RESTART
4 #
5 # FUNCTIONAL DESCRIPTION
6 #
7 # 1. CHECK IF NEW PROGRAM ALLOWED.  IF BIT 1 OF FLAGWRD2 (NODOFLAG) IS SET, AN ALARM 1520 IS CALLED.
8 # 2. CHECK FOR VALIDITY OF PROGRAM SELECTED.  IF AN INVALID PROGRAM IS SELECTED, THE OPERATOR ERROR LIGHT IS
9 #    SET AND CURRENT ACTIVITY, IF ANY, CONTINUE.
10 # 3. SERVICER IS TERMINATED IF IT HAS BEEN RUNNING.
11 # 4. INSTALL IS EXECUTED TO AVOID INTERRUPTING INTEGRATION.
12 # 5. THE ENGINE IS TURNED OFF AND THE DAP IS INITIALIZED FOR COAST.
13 # 6. TRACK AND UPDATE FLAGS ARE SET TO ZERO.
14 # 7. DISPLAY SYSTEM IS RELEASED.
15 # 8. THE FOLLOWING ARE PERFORMED FOR EACH OF THE THREE CASES.
16 #    A. PROGRAM SELECTED IS P00
17 #        1. RENDEZVOUS AND P25 FLAGS ARE RESET.  (KILL P20 AND P25)
18 #        2. STATINT1 IS SCHEDULED BY SETTING RESTART GROUP 2.
19 #        3. MAJOR MODE 00 IS STORED IN THE MODE REGISTER (MODREG).
20 #        4. SUPERBANK 3 IS SELECTED.
21 #        5. NODOFLAG IS RESET.
22 #        6. ALL RESTART GROUPS EXCEPT GROUP2 ARE CLEARED. CONTROL IS TRANSFERRED TO RESTART PROGRAM (GOPROG2)
23 #          WHICH CAUSES ALL CURRENT ACTIVITY TO BE DISCONTINUED AND A 9 MINUTE INTEGRATION CYCLE TO BE
24 #          INITIATED.
25 #    B. PROGRAM SELECTED IS P20 OR P25.
26 #        1. IF THE CURRENT MAJOR MODE IS THE SAME AS THE SELECTED NEWPROGRAM.  THE PROGRAM IS RE-INITIALIZED
27 #          VIA V37XEQ, ALL RESTART GROUPS, EXCEPT GROUP 4 ARE CLEARED.
28 #        2. IF THE CURRENT MAJOR MODE IS NOT EQUAL TO THE NEW REQUEST, A CHECK IS MADE TO SEE IF THE REQUEST-
29 #          ED MAJOR MODE HAS BEEN RUNNING THE BACKGROUND,
30 #          AND IF IT HAS, NO NEW PROGRAM IS SCHEDULED, THE EXISTING
31 #          P20 OR P25 IS RESTARTED TO CONTINUE, AND ITS MM IS SET.
32 #        3. CONTROL IS TRANSFERRED TO GOPROG2.
33 #    C. PROGRAM SELECTED IS NEITHER P00, P20, NOR P25
34 #        1. V37XEQ IS SCHEDULED (AS A JOB) BY SETTING RESTART GROUP 4
35 #        2. ALL CURRENT ACTIVITY EXCEPT RENDEZVOUS AND TRACKING IS DISCONTINUED BY CLEARING ALL RESTART
36 #          GROUPS.  IF THE RENDEZVOUS OR THE P25 FLAG IS ON, GROUP 2 IS NOT CLEARED, ALLOWING THESE PROGRAMS
37 #          TO CONTINUE.
38 #
39 # INPUT/OUTPUT INFORMATION
40 #
41 #    A. CALLING SEQUENCE
42 #        CONTROL IS DIRECTED TO V37 BY THE VERBFAN ROUTINE.
43 #        VERBFAN GOES TO C(VERBTAB+C(VERBREG)).  VERB 37 = MMCHANG.
44 #        MMCHANG EXECUTES A 'TC POSTJUMP', CADR V37.
45 #
46 #    B. ERASABLE INITIALIZATION          NONE
47 #
48 #    C. OUTPUT
```

MAJOR MOD CHANGE

D. DEBRIS

MMNUMBER, MPAC +1, MINDEX, BASETEMP +C(MINDEX), FLAGWRD0, FLAGWRD1, FLAGWRD2, MODREG, GOLOC -1,
GOLOC, GOLOC +1, GOLOC +2, BASETEMP, -PHASE2, PHASE2, -PHASE4

PROGRAM ANALYSIS

A. SUBROUTINES CALLED

ALARM, RELDSP, PINBRNCH, INTSTALL, ENGINOF2, ALLCOAST, V37KLEAN, GOPROG2, FALTON, FINDVAC, SUPERSW,
DSPMM

B. NORMAL EXIT

TC ENDOFJOB

C. ALARMS

1520 (MAJOR MODE CHANGE NOT PERMITTED)

V37

TS MMNUMBER

SAVE MAJOR MODE

CAF PRI030

RESTART AT PINBALL PRIORITY

TS RESTREG

CA IMODES30

IS IMU BEING INITIALIZED

MASK BIT6

CCS A

TCF CANTROD

CS MMNUMBER

IS P70 REQUESTED?

AD DEC70

EXTEND

BZF SETUP70

YES

AD ONE

IS P71 REQUESTED?

EXTEND

BZF SETUP71

YES

CA MMNUMBER

IS NEW REQUEST P00

EXTEND

BZF ISSERVON

YES, CHECK SERVICER STATUS

CS FLAGWRD2

NO, IS NODO V37 FLAG SET

MASK NODOBIT

CCS A

TCF CHECKTAB

NO

CANTROD

TC ALARM

OCT 1520

V37BAD

TC RELDSP

RELEASES DISPLAY FROM ASTRONAUT

TC POSTJUMP

BRING BACK LAST NORMAL DISPLAY IF THERE

CADR PINBRNCH

WAS ONE. OTHERWISE DO AN EOJ.

CHECKTAB

CA NOV37MM

INDEX FOR MM TABLES.

AGAINMM	TS	MPAC +1	
	NDX	MPAC +1	
	CA	PREMM1	# OBTAIN WHICH MM THIS IS FOR
	MASK	LOW7	
	COM		
	AD	MMNUMBER	
	CCS	A	
	CCS	MPAC +1	# IF GR, SEE IF ANY MORE IN LIST
	TCF	AGAINMM	# YES, GET NEXT ONE
	TCF	V37NONO	# LAST TIME OR PASSED MM
	CA	MPAC +1	
	TS	MINDEX	# SAVE INDEX FOR LATER
ISSERVON	CS	FLAGWRD7	# V37 FLAG SET -- I.E., IS SERVICER GOING
	MASK	V37FLBIT	
	CCS	A	
	TCF	CANV37	# NO
	TC	DOWNFLAG	# YES, TURN OFF THE AVERAGE FLAG AND
	ADRES	AVEGFLAG	# WAIT FOR SERVICER TO RETURN TO CANV37
	CAF	V37RETAD	
	TS	OUTROUTE	
	TCF	ENDOFJOB	
V37RET	CS	FLAGWRD0	# IS P20 OR P22 RUNNING?
	MASK	RNDVZBIT	
	CCS	A	
	TCF	+2	# NO. CHECK FOR P25.
	TCF	2.7SPT	# YES. DO 2.7SPOT
	CS	FLAGWRD0	# IS P25 RUNNING?
	MASK	P25FLBIT	
	CCS	A	
2.0SPT	CA	OCT37667	
2.11SPT	AD	BIT5	
2.7SPT	AD	OCT40072	
	TC	PHSCHNGA	
CANV37	CAF	ZERO	
	EXTEND		
	WRITE	SUPERBNK	
	CAF	ROOAD	
	TS	TEMPFLSH	
	TC	PHASCHNG	
	OCT	14	

1				
2				
3	R00	TC	INTPRET	
4				
5		CALL		# WAIT FOR INTEGRATION TO FINISH
6			INTSTALL	
7	DUMMYAD	EXIT		
8				
9		TC	DOWNFLAG	
10		ADRES	3AXISFLG	# RESET 3-AXIT FLAG
11				
12		CAF	LRBYBIT	# CLEAN UP THE R12 FLAGWORD.
13		TS	FLGWRD11	
14				
15		TC	DOWNFLAG	# INSURE THAT THE R04FLAG IS CLEAR.
16		ADRES	R04FLAG	
17				
18		TC	DOWNFLAG	# INSURE MUNFLAG IS CLEAR.
19		ADRES	MUNFLAG	
20				
21		TC	DOWNFLAG	# ALLOW X-AXIS OVERRIDE.
22		ADRES	XOVINFLG	
23		CCS	MMNUMBER	# IS THIS A POOH REQUEST
24		TCF	NOUVEAU	# NO, PICK UP NEW PROGRAM
25				
26	POOH	TC	RELDSP	# RELEASE DISPLAY SYSTEM
27				
28		CAF	PRI05	# SET VARIABLE RESTART PRIORITY FOR
29		TS	PHSPRDT2	# P00 INTEGRATION.
30				
31		TC	CLRADM0D	# CLRADM0D DOES AN INHINT.
32				
33		CS	NODOBIT	# TURN OFF NODOFLAG.
34		MASK	FLAGWRD2	
35		TS	FLAGWRD2	
36				
37		CA	FIVE	# SET RESTART FOR STATEINT1
38		TS	L	
39		COM		
40		DXCH	-PHASE2	
41				
42		CS	OCT700	# TURN OFF P20, P25, IMU IN USE FLAG
43		MASK	FLAGWRD0	
44		TS	FLAGWRD0	# REMDFLG
45				
46		CAF	DNLADP00	
47				
48	SEUDOP00	TS	DNLSTCOD	# SET UP APPROPRIATE DOWNLIST CODE
49		TS	AGSWORD	# (CURRENT LIST WILL BE COMPLETED BEFORE
50				# NEW ONE IS STARTED)
51		TC	IBNKCALL	
52		CADR	ENGNOF1	
53				
54				
55				
56				
57				
58				
59				
60				

	TC	IBNKCALL	# INSURE ALLCOAST.
	CADR	ALLCOAST	# DOES A RESTORDB.
	CS	OCT120	# TURN OFF TRACK, UPDATE FLAGS
	TS	EBANKTEM	
	MASK	FLAGWRD1	
	TS	FLAGWRD1	
	TC	IBNKCALL	# KILL GROUPS 1,3,5,6
	CADR	V37KLEAN	
	CCS	MMNUMBER	# IS IT POOH
	TCF	RENDV00	# NO
GOMOD	TC	IBNKCALL	# REDUNDANT EXCEPT FOR GROUP 4
	CADR	POOKLEAN	
	CA	MMNUMBER	
	TS	MODREG	
GOGOPROG	TC	POSTJUMP	
	CADR	GOPROG2	
RENDV00	CS	MODREG	# IS CURRENT PROGRAM 22
	AD	OCT26	
	EXTEND		
	BZF	RESET22	# YES -- CLEAR RENDEZVOUS FLAG
	CS	MMNUMBER	# IS NE PROGRAM P22
	AD	OCT26	
	EXTEND		
	BZF	RESET22	
	AD	NEG2	# IS NEW PROGRAM = P20 OR P25
	EXTEND		
	BZF	RENDN00	# YES
	AD	FIVE	# 25
	EXTEND		
	BZF	RENDN00	# YES
	CA	OCT500	# NO, IS EITHER P20 OR P25 RUNNING
	MASK	FLAGWRD0	
	CCS	A	
	TCF	P00FIZZ	# YES, LEAVE GROUP 2 TO PICK UP P20 OR P25
RESET22	CS	OCT700	# CLEAR RENDEZVOUS, P25
	MASK	FLAGWRD0	# AND IMU IN USE FLAGS
	TS	FLAGWRD0	
	TC	CLRADM0D	

KILL2	EXTEND		# NO, KILL 2
	DCA	NEGO	
	DXCH	-PHASE2	
POOFIZZ	CAF	V37QCAD	# RESTART POINT FOR V37XEQ
	TS	TEMPFLSH	
	TCF	GOGOPROG	
RENDN00	CS	MODREG	
	AD	OCT24	
	EXTEND		
	BZF	KILL2	# P20 OR P25 ON TOP OF P20 OR P25 --
	AD	FIVE	
	EXTEND		
	BZF	KILL2	
	CA	OCT500	
	MASK	FLAGWRD0	
	AD	MMNUMBER	
	COM		
	AD	P20REG	# IS IT 20 AND IS RENDEZVOUS FLAG ON
	EXTEND		
	BZF	STATQUO	# YES
	AD	OCT305	# IS IT 25 AND IS P25 BIT ON
	EXTEND		
	BZF	STATQUO	# YES, LEAVE AS IS
	TCF	KILL2	
STATQUO	CS	FLAGWRD1	# SET TRACKFLAG
	MASK	OCT120	#
	ADS	FLAGWRD1	UPDATE FLAG
	TCF	GOMOD	
NOUVEAU	CAF	OCT500	# IS P20 OR P25 FLAG SET
	MASK	FLAGWRD0	
	CCS	A	
	TCF	+3	# YES
	TC	DOWNFLAG	# NO, RESET IMUINUSE FLAG
	ADRES	IMUSE	
	INDEX	MINDEX	
	CAF	DNLADMM1	# OBTAIN APPROPRIATE DOWNLIST ADDRESS
	INHINT		
	TCF	SEUDOP00	
V37NONO	TC	FALTON	# COME HERE IF MM REQUESTED DOESN'T EXIST

	TCF	V37BAD	
OCT00010	EQUALS	BIT4	
OCT500	OCT	500	# BITS 7 AND 9
OCT305	OCT	305	
OCT26	OCT	26	
P20REG	OCT	124	
V37XEQ	INHINT		
	INDEX	MINDEX	
	CAF	PREMM1	# OBTAIN PRIO, EBANK, AND MM
	TS	MMTEMP	
	TS	CYR	# SHIFT RIGHT TO BITS 14-10
	CA	CYR	
	MASK	PRI037	
	TS	PHSPRDT4	# PRESET GROUP 4 RESTART PRIORITY
	TS	NEWPRIO	# STORE PRIO FOR SPVAC
	CA	MMTEMP	# OBTAIN EBANK -- BITS 8, 9, 10 OF MMTEMP.
	EXTEND		
	MP	BIT8	
	MASK	LOW3	
	TS	L	
	INDEX	MINDEX	
	CAF	FCADRMM1	
	TS	BASETEMP	# MAKE BBCON BY ADDING HI5 OR FCADR
	MASK	HI5	
	ADS	L	
	CA	BASETEMP	# OBTAIN GENADR PORTION OF 2CADR.
	MASK	LOW10	
	AD	BIT11	
	TC	SPVAC	
V37XEQC	CA	MMTEMP	# UPON RETURN FROM FINDVAC PLACE THE
	MASK	LOW7	# NEW MM IN MODREG (THE LOW 7 BITS OF
	TC	NEWMODEA	# PHSPRDT1)
	TC	RELDSP	# RELEASE DISPLAY
	TC	ENDOFJOB	# AND EXIT
NEG7	EQUALS	OCT77770	
MMTEMP	EQUALS	PHSPRDT3	
BASETEMP	EQUALS	TBASE4	
V37QCAD	CADR	V37XEQ +3	
R00AD	CADR	DUMMYAD	

V37RETAD	CADR	V37RET
OCT37667	OCT	37667
OCT40072	OCT	40072
OCT700	OCT	700

SETUP71	CAF	THREE
SETUP70	TS	Q
	EXTEND	
	DCA	P70CADR
	AD	Q
	DTCB	

DEC70	DEC	70
	EBANK=	R
P70CADR	2CADR	P70

FOR VERB 37 TWO TABLES ARE MAINTAINED. EACH TABLE HAS AN ETRY FOR EACH
MAJOR MODE THAT CAN BE STARTED FROM THE KEYBOARD. THE ENTRIES ARE PUT
INTO THE TABLE WITH THE ENTRY FOR THE HIGHEST MAJOR MODE COMING FIRST,
TO THE LOWEST MAJOR MODE WHICH IS THE LAST ENTRY IN EACH TABLE.

THE FCADRM TABLE CONTAINS THE FCADR OF THE STARTING JOB OF
THE MAJOR MODE. FOR EXAMPLE,

#	#	#	#
#	FCADRM1	FCADR	P79
#		FCADR	PROG18
#		FCADR	P01
#			

NOTE: THE FIRST ENTRY MUST BE LABELED FCADRM1.

FCADRM1	FCADR	P79
	FCADR	P78
	FCADR	P76
	FCADR	P75
	FCADR	P74
	FCADR	P73
	FCADR	P72
	FCADR	LANDJUNK
	FCADR	P63LM
	FCADR	P57
	FCADR	PROG52
	FCADR	P51
	FCADR	P47LM
	FCADR	P42LM
	FCADR	P41LM
	FCADR	P40LM
	FCADR	P39
	FCADR	P38

```
FCADR P35
FCADR P34
FCADR P33
FCADR P32
FCADR P31
FCADR P30
FCADR PROG25
FCADR PROG22
FCADR PROG21
FCADR PROG20
FCADR P12LM
FCADR P06
```

THE PREMM TABLE CONTAINS THE E-BANK, MAJOR MODE, AND PRIORITY
INFORMATION, IT IS IN THE FOLLOWING FORM,

PPP PPE EEM MMM MMM

WHERE THE 7 M BITS CONTAIN THE MAJOR MODE NUMBER
3 E BITS CONTAIN THE E-BANK NUMBER
5 P BITS CONTAIN THE PRIORITY AT WHICH THE JOB IS
TO BE STARTED

FOR EXAMPLE,

#	PREMM1	OCT	67213	# PRIORITY	33
#				# E-BANK	5
#				# MAJOR MODE	11
#		OCT	25437	# PRIORITY	12
#				# E-BANK	6
#				# MAJOR MODE	31

NOTE: THE FIRST ENTRY MUST BE LABELED PREMM1

PREMM1	OCT	27717	# MM 79	EBANK 7	PRI0 13
	OCT	27716	# MM 78	EBANK 7	PRI0 13
	OCT	27714	# MM 76	EBANK 7	PRI0 13
	OCT	27713	# MM 75	EBANK 7	PRI0 13
	OCT	27712	# MM 74	EBANK 7	PRI0 13
	OCT	27711	# MM 73	EBANK 7	PRI0 13
	OCT	27710	# MM 72	EBANK 7	PRI0 13
	OCT	27704	# MM 68	EBANK 7	PRI0 13
	OCT	27677	# MM 63	EBANK 7	PRI0 13
	OCT	27271	# MM 57	EBANK 5	PRI0 13
	OCT	27264	# MM 52	EBANK 5	PRI0 13
	OCT	27263	# MM 51	EBANK 5	PRI0 13
	OCT	27657	# MM 47	EBANK 7	PRI0 13
	OCT	27652	# MM 42	EBANK 7	PRI0 13
	OCT	27651	# MM 41	EBANK 7	PRI0 13
	OCT	27650	# MM 40	EBANK 7	PRI0 13
	OCT	27647	# MM 39	EBANK 7	PRI0 13
	OCT	27646	# MM 38	EBANK 7	PRI0 13

OCT	27643	# MM 35	EBANK 7	PRI0 13
OCT	27642	# MM 34	EBANK 7	PRI0 13
OCT	27641	# MM 33	EBANK 7	PRI0 13
OCT	27640	# MM 32	EBANK 7	PRI0 13
OCT	27637	#		
OCT	27636	# MM 30	EBANK 7	PRI0 13
OCT	27631	# MM 25	EBANK 7	PRI0 13
OCT	27626	# MM 22	EBANK 7	PRI0 13
OCT	27625	# MM 21	EBANK 7	PRI0 13
OCT	27624	# MM 20	EBANK 7	PRI0 13
OCT	27614	# MM 12	EBANK 7	PRI0 13
OCT	27006	# MM 06	EBANK 4	PRI0 13

NOTE: THE FOLLOWING CONSTANT IS THE NUMBER OF ENTRIES IN EACH OF
----- THE ABOVE LISTS-1 (I.E., THE NUMBER OF MAJOR MODES (EXCEPT P00)
THAT CAN BE CALLED FROM THE KEYBOARD MINUS ONE)

NOV37MM DEC 29 # MM'S -1

DNLADMM1 ADRES RENDEZVU # P79

ADRES RENDEZVU # P78

ADRES RENDEZVU

ADRES RENDEZVU # P75

ADRES RENDEZVU # P74

ADRES RENDEZVU # P73

ADRES RENDEZVU # P72

ADRES DESASCNT # P68

ADRES DESASCNT # P63

ADRES LUNRSALN # P57

ADRES COSTALIN # P52

ADRES COSTALIN # P51

ADRES ORBMANUV # P47

ADRES ORBMANUV # P42

ADRES ORBMANUV # P41

ADRES ORBMANUV # P40

ADRES RENDEZVU # P39

ADRES RENDEZVU # P38

ADRES RENDEZVU # P35

ADRES RENDEZVU # P34

ADRES RENDEZVU # P33

ADRES RENDEZVU # P32

ADRES RENDEZVU # P31LM

ADRES RENDEZVU # P30

ADRES RENDEZVU # P25

ADRES LUNRSALN # P22

ADRES RENDEZVU # P21

ADRES RENDEZVU # P20

ADRES DESASCNT # P12

ADRES COSTALIN # P06

DNLADP00 = ZERO

COSTALIN = 0

FRESH_START_AND_RESTART

1					1
2	AGSUPDAT	=	1		2
3	RENDEZVU	=	2		3
4	ORBMANUV	=	3		4
5	DESASCNT	=	4		5
6	LUNRSALN	=	5		6
7					7
8		BANK	13		8
9		SETLOC	INTINIT		9
10		BANK			10
11					11
12		COUNT*	\$\$/INTIN		12
13					13
14		EBANK=	RRECTCSM		14
15					15
16	# THIS ROUTINE DOES THE P00 INTEGRATION				16
17					17
18	STATEUP	SET	BOF	# EXTRAPOLATE CM STATE VECTOR	18
19					19
20			VINTFLAG	# ALSO 6X6 W-MATRIX IF LM ON LUNAR	20
21			SURFFLAG	# SURFACE AND W-MATRIX VALID	21
22			DOINT	#	22
23		BOF	SET	# FOR RENDEZVOUS NAVIGATION.	23
24					24
25			RENDWFLG		25
26			DOINT		26
27	DOINT	CLEAR	DIMOFLAG		27
28			CALL		28
29			PRECIFLG	# ENGAGES 4-TIME STEP LOGIC IN INTEGRATION	29
30			INTEGRV	# WHEN MODREG = 0	30
31					31
32		BON	DLOAD		32
33					33
34			SURFFLAG		34
35			NO-INT		35
36			TETCSM		36
37		STCALL	TDEC1		37
38			INTSTALL		38
39		CLEAR	CALL	# EXTRAPOLATE LM STATE VECTOR	39
40					40
41			VINTFLAG		41
42			SETIFLGS		42
43		BOF		# ALSO 9X9 W-MATRIX IF W IS VALID	43
44					44
45			RENDWFLG		45
46			DOINT2		46
47		SET	SET		47
48					48
49			DIMOFLAG		49
50			D6OR9FLG		50
51	DOINT2	SET	CALL		51
52					52
53			PRECIFLG	# DISENGAGE 4 TIME STEP LOGIC IN INTEG.	53
54			INTEGRV		54
55	NO-INT	CLRG0			55
56					56
57			NODOFLAG		57
58			ENDINT		58
59					59
60					60



1412THE

FRESH_START_AND_RESTART

PAGE 237

1				1
2	# THISVINT IS CALLED BY MIDTOAV1 AND 2			2
3				3
4	THISVINT	CLEAR	RVQ	4
5			VINTFLAG	5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

#

```
# THERE ARE TWO FORMS OF RESTART TABLES FOR EACH GROUP.  THEY ARE KNOWN AS THE EVEN RESTART TABLES AND THE ODD
# RESTART TABLES.  THE ODD TABLES HAVE ONLY ONE ENTRY OF THREE LOCATIONS WHILE THE EVEN TABLES HAVE TWO ENTRIES
# EACH USING THREE LOCATIONS.  THE INFORMATION AS TO WHETHER IT IS A JOB, WAITLIST, OR A LONGCALL IS GIVEN BY THE
# WAY THINGS ARE PUT INTO THE TABLES.
```

```
# A JOB HAS ITS PRIORITY STORED IN PRDRTAB OF THE CORRECT PHASE SPOT - A POSITIVE PRIORITY INDICATES A
# FINDVAC JOB, A NEGATIVE PRIORITY A NOVAC. THE 2CADR OF THE JOB IS STORED IN THE CADRTAB.
# FOR EXAMPLE,
```

#	5.7SPOT	OCT	23000
#		2CADR	SOMEJOB
#			

```
# A RESTART OF GROUP 5 WITH PHASE SEVEN WOULD THEN CAUSE SOMEJOB TO BE RESTARTED AS A FINDVAC WITH PRIORITY 23.
```

#	5.5SPOT	OCT	-23000
#		2CADR	ANYJOB

HERE A RESTART OF GROUP 5 WITH PHASE 7 WOULD CAUSE ANYJOB TO BE RESTARTED AS A NOVAC WITH PRIORITY 23.

```
# A LONGCALL HAS ITS GENADR OF ITS 2CADR STORED NEGATIVELY AND ITS BBCON STORED POSITIVELY.  IN ITS PRDTTAB IS
# PLACED THE LOCATION OF A DP REGISTER THAT CONTAINS THE DELTA TIME THAT LONGCALL HAD BEEN ORIGINALLY STARTED
# WITH.  EXAMPLE,
```

# 3.6SPOT	GENADR	DELTAT
#	-GENADR	LONGTASK
#	BBCON	LONGTASK

#	OCT	31000
#	2CADR	JOBAGAIN

```
# THIS WOULD START UP LONGTASK AT THE APPROPRIATE TIME, OR IMMEDIATELY IF THE TIME HAD ALREADY PASSED. IT SHOULD
# BE NOTED THAT IF DELTAT IS IN A SWITCHED E BANK, THIS INFORMATOIN SHOULD BE IN THE BBCON OFTHE 2CADR OF THE
# TASK. FROM ABOVE, WE SEE THAT THE SECOND PART OF THIS PHASE WOULD BE STARTED AS A JOB WITH A PRIORITY OF 31.
#
```

```
# WAITLIST CALLS ARE IDENTIFIED BY THE FACT THAT THEIR 2CADR IS STORED NEGATIVELY. IF PRDTTAB OF THE PHASE SPOT
# IS POSITIVE, THEN IT CONTAINS THE DELTA TIME, IF PRDTTAB IS NEGATIVE THEN IT IS THE -GENADR OF AN ERASABLE
# LOCATION CONTAINING THE DELTA TIME, THAT IS, THE TIME IS STORED INDIRECTLY. IT SHOULD BE NOTED AS ABOVE, THAT
# IF THE TIME IS STORED INDIRECTLY, THE BBCON MUST CONTAIN THE NECESSARY E BANK INFORMATION IF APPLICABLE. WITH
# WAITLIST WE HAVE ONE FURTHER OPTION, IF -0 IS STORED IN PRDTTAB, IT WILL CAUSE AN IMMEDIATE RESTART OF THE
# TASK.  EXAMPLES,
```

```
#          OCT      77777          # THIS WILL CAUSE AN IMMEDIATE RESTART
#      -2CADR    ATASK          # OF THE TASK :ATASK:
```

#				# IF THE TIME OF THE 2 SECONDS SINCE DUMMY
#	DEC	200		# WAS PUT ON THE WAITLIST IS UP, IT WILL BEGIN
#	-2CADR	DUMMY		# IN 10 MS, OTHERWISE IT WILL BEGIN WHEN
#				# IT NORMALLY WOULD HAVE BEGUN.


```
1
2 # -GENADR DTIME # WHERE DTIME CONTAINS THE DELTA TIME
3 # -2CADR TASKTASK # OTHERWISE THIS IS AS ABOVE
4 #
5 # ***** NOW THE TABLES THEMSELVES *****
6
7 BANK 01
8 SETLOC RESTART
9 BANK
10
11 PRDTTAB EQUALS 12000 # USED TO FIND THE PRIORITY OR DELTATIME
12 CADRTAB EQUALS 12001 # THIS AND THE NEXT RELATIVE LOC CONTAIN
13 # RESTART 2CADR
14
15 COUNT* $$/RSTAB # TABLES IN BANK 1.
16 SIZETAB TC 1.2SPOT -12006
17 TC 1.3SPOT -12004
18 TC 2.2SPOT -12006
19 TC 2.3SPOT -12004
20 TC 3.2SPOT -12006
21 TC 3.3SPOT -12004
22 TC 4.2SPOT -12006
23 TC 4.3SPOT -12004
24 TC 5.2SPOT -12006
25 TC 5.3SPOT -12004
26 TC 6.2SPOT -12006
27 TC 6.3SPOT -12004
28 1.2SPOT OCT 21000 # A DUMMY EXAMPLE TO BE REPLACED AS SOON
29 EBANK= STATE
30 2CADR ENDOFJOB # AS THERE IS A LEGITIMATE 1.2SPOT
31
32 DEC 100
33 EBANK= STATE
34 2CADR TASKOVER
35
36 # ANY MORE GROUP 1.EVEN RESTART VALUES SHOULD GO HERE
37
38 1.3SPOT -GENADR SAVET-30
39 EBANK= DVCNTR
40 -2CADR ULLGTASK
41
42 # ANY MORE GROUP 1.ODD RESTART VALUES SHOULD GO HERE
43
44 2.2SPOT EQUALS 1.2SPOT
45 # ANY MORE GROUP 2.EVEN RESTART VALUES SHOULD GO HERE
46
47 2.3SPOT GENADR 600SECS
48 -GENADR STATEINT
49 EBANK= RRECTCSM
50 BBCON STATEINT
51
52
53
54
55
56
57
58
59
60
```

RESTART_TABLES

1				1
2	2.5SPOT	OCT	05000	2
3		EBANK=	RRECTCSM	3
4		2CADR	STATINT1	4
5				5
6	2.7SPOT	DEC	1500	6
7		EBANK=	LOSCOUNT	7
8		-2CADR	P20LEMC1	8
9				9
10	2.11SPOT	OCT	14000	10
11		EBANK=	P21TIME	11
12		2CADR	P25LEM1	12
13				13
14	2.13SPOT	OCT	10000	14
15		EBANK=	LOSCOUNT	15
16		2CADR	RELINUS	16
17				17
18	2.15SPOT	OCT	26000	18
19		EBANK=	LOSCOUNT	19
20		2CADR	R22RSTRT	20
21				21
22	2.17SPOT	OCT	77777	22
23		EBANK=	VGPREV	23
24		-2CADR	RED02.17	24
25				25
26	2.21SPOT	DEC	25	26
27		EBANK=	DVCNTR	27
28		-2CADR	R10,R11	28
29				29
30	# ANY MORE GROUP 2.ODD RESTART VALUES SHOULD GO HERE.			30
31				31
32	3.2SPOT	EQUALS	1.2SPOT	32
33	# ANY MORE GROUP 3.EVEN RESTART VALUES SHOULD GO HERE			33
34				34
35	3.3SPOT	-GENADR	ZOOMTIME	35
36		EBANK=	DVCNTR	36
37		-2CADR	ZOOM	37
38				38
39	3.5SPOT	OCT	20000	39
40		EBANK=	TTOGO	40
41		2CADR	S40.13	41
42				42
43	# ANY MORE GROUP 3.ODD RESTART VALUES SHOULD GO HERE			43
44				44
45	4.2SPOT	DEC	2500	45
46		EBANK=	TTOGO	46
47		-2CADR	TIG-5	47
48				48
49		OCT	77777	49
50		EBANK=	TTOGO	50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1				1
2	-2CADR	RED04.2		2
3				3
4	# ANY MORE GROUP 4.EVEN RESTART VALUES SHOULD GO HERE			4
5				5
6	4.3SPOT	OCT	25000	6
7		EBANK=	DVCNTR	7
8		2CADR	GOABORT	8
9				9
10	4.5SPOT	DEC	50	10
11		EBANK=	TTOGO	11
12		-2CADR	ULLAGOFF	12
13				13
14	4.7SPOT	DEC	500	14
15		EBANK=	DVCNTR	15
16		-2CADR	TIG-0	16
17				17
18	4.11SPOT	-GENADR	TGO +1	18
19		EBANK=	DVCNTR	19
20		-2CADR	ENGOFTSK	20
21				21
22	4.13SPOT	OCT	12000	22
23		EBANK=	TRKMKCNT	23
24		2CADR	POSTBURN	24
25				25
26	4.15SPOT	DEC	500	26
27		EBANK=	TTOGO	27
28		-2CADR	TIG-30	28
29				29
30	4.17SPOT	OCT	77777	30
31		EBANK=	DVCNTR	31
32		-2CADR	TIG-5	32
33				33
34	4.21SPOT	OCT	13000	34
35		EBANK=	STAR	35
36		2CADR	R51.1 +1	36
37				37
38	4.23SPOT	OCT	77777	38
39		EBANK=	DVCNTR	39
40		-2CADR	IGNITION	40
41				41
42	4.25SPOT	GENADR	SAVET-30	42
43		-GENADR	TIG-35	43
44		EBANK=	SAVET-30	44
45		BBCON	TIG-35	45
46				46
47	4.27SPOT	OCT	52777	47
48		EBANK=	DVCNTR	48
49		2CADR	P70A	49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

RESTART_TABLES

4.31SPOT OCT 52777
EBANK= DVCNTR
2CADR P71A

4.33SPOT OCT 46777
EBANK= DVCNTR
2CADR GOP00FIX

4.35SPOT OCT 46777
EBANK= DVCNTR
2CADR GOP00D00

4.37SPOT OCT 52777
EBANK= WHICH
2CADR COMFAIL

ANY MORE 4.ODD RESTART VALUES SHOULD GO HERE.

5.2SPOT OCT 22000
EBANK= DVCNTR
2CADR NORMLIZE

DEC 200
EBANK= DVCNTR
-2CADR REREADAC

5.4SPOT DEC 200
EBANK= DVCNTR
-2CADR REREADAC

OCT 20000
EBANK= DVCNTR
2CADR SERVICER

ANY MORE GROUP 5.EVEN RESTART VALUES SHOULD GO HERE

5.3SPOT DEC 200
EBANK= DVCNTR
-2CADR REREADAC

5.5SPOT OCT 77777
EBANK= DVCNTR
-2CADR RED05.5

5.7SPOT OCT 77777
EBANK= DVCNTR



1					1
2	-2CADR BIBIBIAS				2
3					3
4	# ANY MORE GROUP 5.ODD RESTART VALUES SHOULD GO HERE				4
5					5
6	6.2SPOT	EQUALS	1.2SPOT		6
7	6.3SPOT	DEC	100		7
8		EBANK=	TIG		8
9		-2CADR	CLOKTASK		9
10					10
11	6.5SPOT	OCT	30000	# PROTECT INCREMENTING OF TIME2,TIME1 BY	11
12		EBANK=	TEPHEM	# P27(UPDATE PROGRAM) VIA V70 OR V73.	12
13		2CADR	TIMEDIDR		13
14					14
15	6.7SPOT	OCT	17000		15
16		EBANK=	VGPREV		16
17		2CADR	RED06.7		17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1				
2		BANK	12	
3		SETLOC	AOTMARK1	
4		BANK		
5				
6		EBANK=	XYMARK	
7		COUNT*	\$\$/MARK	
8				
9	AOTMARK	INHINT		
10		CCS	MARKSTAT	# SEE IF AOTMARK BUSY
11		TC	+2	# MARK SYSTEM BUSY -- DO ALARM
12		TC	EXTVBCHK	
13		TC	POOD00	
14		OCT	00105	
15				
16	EXTVBCHK	CAF	SIX	# SEE IF EXT. VERB WORKING
17		MASK	EXTVBACT	
18		CCS	A	
19		TCF	MKABORT	# YES -- ABORT
20				
21		CAF	BIT2	# NO -- DISALLOW SOME EXTENDED VERB ACTION
22		ADS	EXTVBACT	# BIT2 RESET IN ENDMARK
23	MKVAC	CCS	VAC1USE	# LOOK FOR A VAC AREAD -- DO ABORT IF
24		TCF	MKVACFND	# NONE AVAILABLE
25		CCS	VAC2USE	
26		TCF	MKVACFND	
27		CCS	VAC3USE	
28		TCF	MKVACFND	
29		CCS	VAC4USE	
30		TCF	MKVACFND	
31		CCS	VAC5USE	
32		TCF	MKVACFND	
33		DXCH	BUF2	
34		TC	BAILOUT1	# ALL VAC AREAS OCCUPIED -- ABORT.
35		OCT	01207	
36				
37	MKVACFND	AD	TWO	
38		TS	MARKSTAT	# STORE VAC ADR IN LOW 9 OF MARKSTAT
39				
40		CAF	ZERO	
41		INDEX	MARKSTAT	
42		TS	0 -1	# ZERO IN VACUSE REG TO SHOW VAC OCCUPIED
43				
44		CAF	PRI015	
45		TC	FINDVAC	# SET UP JOB FOR GETDAT
46		EBANK=	XYMARK	
47		2CADR	GETDAT	
48				
49		RELINT		
50		TCF	SWRETURN	
51				
52				
53				
54				
55				
56				
57				
58				
59				
60				

MKRELEAS

KILLAOT	CAF TS	ZERO EXTVBACT	# TERMINATE AOTMARK -- ALLOW EXT VERB
GETDAT	TC CS MASK	GOTOPOOH MARKSTAT BIT12	# SET BIT12 TO DISCOURAGE MARKRUPT # BIT12 RESET AT GETMARK
	ADS	MARKSTAT	
	CAF TC CADR	V01N71 BANKCALL GOMARKF	# DISPLAY DETENT AND STAR CODE
ENTERDAT	TCF TCF TCF	KILLAOT DODAT GETDAT	# V34 -- DOES GOTOPOOH # V33 -- PROCEED -- USE THIS STAR FOR MARKS # ENTER -- REDISPLAY STAR CODE
DODAT	CAF MASK	HIGH9 AOTCODE	# PICK DETENT CODE FROM BITS7-9 OF AOTCODE # AND SEE IF CODE 1 TO 6
	EXTEND MP TS	BIT9 XYMARK	# STORE DETENT
	EXTEND BZMF	GETDAT	# COAS CALIBRATION CODE - NO GOOD HERE
	AD EXTEND BZF	NEG7 CODE7	# SEE IF DETENT 7 FOR COAS
	TCF	CODE1T06	
CODE7	CAF TC CADR	V06N87* BANKCALL GOMARKF	# CODE 7, COAS SIGHTING, GET OPTIC AXIS # AZ AND EL OF SIGHTING DEVICE FROM ASTRO
	TCF TCF TCF EXTEND	KILLAOT +2 CODE7	# V34 -- DOES GOTOPOOH # PROCEED # ON ENTER, RECYCLE
	DCA INDEX DXCH	AZ FIXLOC 8D	# PICK UP AZ AND EL IN SP 25 COMP # STORE IN 8D AND 9D OF LOCAL VAC
	CAF TCF	ZERO COASCODE	# BACKUP SYSTEM TO BE USED # ZERO APPARENT ROTATION
CODE1T06	INDEX CA INDEX TS	XYMARK AOTEL -1 FIXLOC 9D	# INDEX AOT POSITION BY DET CODE # STORE ELEVATION IN VAC+9D
	INDEX	XYMARK	# INDEX DET CODE 1,2 OR 3

1412THE

```

1 # THE OPTAXIS SUBROUTINE COMPUTES THE X AND Y MARK PLANE VECs AND
2 # ROTATES THEM THRU THE APPARENT FIELD OF VIEW ROTATION UNIQUE TO AOT
3 # OPTAXIS USES OANB TO COMPUTE THE OPTIC AXIS
4 #
5 #
6 # INPUT -- AZIMUTH ANGLE IN SINGLE PREC AT CDU SCALE IN 8D OF JOB VAC
7 # ELEVATION ANGLE IN SINGLE PREC AT CDU SCALE IN 9D OF JOB VAC
8 # ROTATION ANGLE IN SINGLE PREC IS COMP SCALED BY PI IN 10D OF VAC
9 #
10 # OUTPUT -- OPTIC AXIS VEC IN NG COORDS IN SCAXIS
11 # X-MARK PLANE 1/4VEC IN NB COORDS AT 18D OF JOB VAC
12 # Y-MARK PLANE 1/4VEC IN NB COORDS AT 12D OF JOB VAC

```

OPTAXIS	CALL		# GO COMPUTE OA AN X AND Y PLANE VECs
		OANB	
	SLOAD	SR1	# LOAD APP ROTATION IN ONES COMP
		10D	# RESCALE BY 2PI
	PUSH	SIN	# 1/2SIN(ROT) 0-1
	PDDL	COS	
	PUSH	VXSC	# 1/2COS(ROT) 2-3
		18D	
	PDDL	VXSC	# 1/4COS(ROT)UYP 4-9
		0	
		24D	# 1/4SIN(ROT)UXP
	BVSU	STADR	# UP 4-9
	STODL	12D	# YPNB=1/4(COS(ROT)UYP-SIN(ROT)UXP)
	VXSC	PDDL	# UP 2-3 UP 0-1 FOR EXCHANGE
		24D	# 1/4COS(ROT)UXP PUSH 0-5
	VXSC	VAD	# 1/4SIN(ROT)UYP
		18D	# UP 0-5
	STADR		
	STOVL	18D	# XPNB=1/4(COS(ROT)UXP+SIN(ROT)UYP)
		LO6ZEROS	# INITIALIZE AVE STAR VEC ACCUMULATOR
	STORE	STARAD +6	
	EXIT		
	TCF	GETMKS	

```
# THE OANB SUBROUTINE COMPUTES THE OPTIC AXIS OF THE SIGHTING INSTRUMENT
# FROM AZIMUTH AND ELEVATION INPUT FROM THE ASTRONAUT.
```

```
#
# INPUT --      AZIMUTH ANGLE IN SINGLE PREC 2'S COMP IN 8D OF JOB VAC
#              ELEVATION ANGLE IN SINGLE PREC 2'S COMP IN 9D OF VAC
```

```
#
# OUTPUT --     OPTIC AXIS IN NB COORDS. IN SCAXIS
#              X-PLANE 1/2VEC IN NB COORDS AT 24D OF VAC
#              Y-PLANE 1/2VEC IN NB COORDS AT 18D OF VAC
```

```
BANK 05
SETLOC AOTMARK2
BANK
```

```
COUNT* $$/MARK
```

```
OANB
```

```
SETPD STQ
```

```
0
```

```
GCTR      # STORE RETURN
```

```
SLOAD
```

```
RTB
```

```
9D      # PICK UP SP ELV
```

```
CDULOGIC
```

```
PUSH
```

```
COS
```

```
PDDL
```

```
SIN
```

```
# 1/2COS(ELV) PD 0-1
```

```
STADR
```

```
STODL
```

```
SCAXIS
```

```
# OAX=1/2SIN(ELV)
```

```
8D
```

```
RTB
```

```
CDULOGIC
```

```
PUSH
```

```
COS
```

```
STORE
```

```
20D
```

```
# STORE UYP(Y) 20-21
```

```
PDDL
```

```
SIN
```

```
# 1/2COS(AZ) PD 2-3
```

```
PUSH
```

```
DCOMP
```

```
# PUSH 1/2S IN (AZ) 4-5
```

```
STODL
```

```
22D
```

```
# STORE UYP(Z) 22-23
```

```
LO6ZEROS
```

```
STODL
```

```
18D
```

```
# STORE UYP(X) 18-19
```

```
DMP
```

```
SL1
```

```
0
```

```
STODL
```

```
SCAXIS +2
```

```
# OAY=1/2COS(ELV)SIN(AZ)
```

```
DMP
```

```
SL1
```

```
# UP 2-3
```

```
STADR
```

```
# UP 0-1
```

```
STOVL
```

```
SCAXIS +4
```

```
# OAZ=1/2COS(ELV)COS(AZ)
```

```
18D
```

```
# LOAD UYP VEC
```

```
VXV
```

```
UNIT
```

```
STORE
```

```
SCAXIS
```

```
# UXP VEC=UYP X OA
```

```
STORE
```

```
24D
```

```
# STORE UXP
```

```
GOTO
```

```
GCTR
```

```
1 # SURFSTAR COMPUTES A STAR VECTOR IN SM COORDINATES FOR LUNAR
2 # SURFACE ALIGNMENT AND EXITS TO AVEIT TO AVERAGE STAR VECTORS.
3
4 #
5 #      GIVEN      X-MARK PLANE 1/4 VEC IN NB AT 18D OF LOCAL VAC
6 #      Y-MARK PLANE 1/4 VEC IN NB AT 12D OF LOCAL VAC
7 #
8 #      CURSOR SP 2COMP AT POSITION 1 OF INDEXED MARKVAC
9 #      SPIRAL SP 2COMP AT POSITION 3 OF INDEXED MARKVAC
10 #      CDUY,Z,X AT POSITIONS 0,2,4 OF INDEXED MARKVAC
11
12      BANK      15
13      SETLOC    P50S
14
15      BANK
16      COUNT*    $$/R59
17
18 SURFSTAR      VLOAD*
19                0,1          # PUT X-MARK CDUS IN CDUSPOT FOR TRG*NBSM
20                STORE        CDUSPOT
21      SLOAD*    RTB
22                1,1          # PICK UP YROT
23                CDULOGIC
24      STORE     24D          # STORE CURSOR FOR SPIRAL COMP (REVS)
25      BZE
26                YZCHK        # IF YROT ZERO -- SEE IF SROT ZERO
27
28 JUSTZY        PUSH        COS
29                PDDL        SIN          # 1/2COS(YROT) 0-1
30                VXSC        PDDL        # UP 0-1      1/8SIN(YROT)UXP 0-5
31                18D
32      VXSC      VSU          # UP      0-5
33                12D          # UYP
34
35      UNIT      VXV
36      UNIT      SCAXIS
37      UNIT      PUSH
38      SLOAD*    RTB
39                3,1          # PICK UP SPIRAL
40                CDULOGIC
41
42      STORE     26D          # STORE SPIRAL (REVS)
43      DSU       DAD
44                24D
45      DMP        ABOUTONE
46                DMP
47                DP1/12
48      STORE     26D          # SEP=(360 + SPIRAL -CURSOR)/12
49      SIN       VXSC          # UP      0-5
50      VSL1      PDDL        # 1/2SIN(SEP)(UPP X OA) 0-5
51                26D
52      COS       VXSC
53                SCAXIS
54
55 JUSTOA        VSL1        VAD          # UP      0-5
56      UNIT      CALL
57                TRG*NBSM
58
59      STCALL    24D          # STAR VEC IN SM
60                AVEIT        # GO AVERAGE
```



AOTMARK

PAGE 251

1412THE

1					1
2	ABOUTONE	2DEC	.99999999		2
3					3
4	DP1/12	EQUALS	DEG30	# .08333333	4
5		BANK	7		5
6		SETLOC	AOTMARK1		6
7					7
8	YZCHK	BANK			8
9		COUNT*	\$\$/MARK		9
10		SLOAD*	BZE	# YROT ZERO AND IF SROT ZERO FORCE STAR	10
11			3,1	# ALONG OPTIC AXIS	11
12		DLOAD	YSZERO		12
13					13
14	YSZERO	VLOAD	24D	# SROT NOT ZERO -- CONTINUE NORMALLY	14
15			JUSTZY		15
16			GOTO		16
17			SCAXIS		17
18			JUSTOA		18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

THE GETMKS ROUTINE INITIALIZES THE SIGHTING MARK PROCEDURE

GETMKS CAF ZERO # INITIALIZE MARK ID REGISTER AND MARK CNT

TS XYMARK
TS MARKCNTR

CAF LOW9 # ZERO BITS10 TO 15 RETAINING MKVAC ADR

MASK MARKSTAT
TS MARKSTAT

PASTIT CAF MKVB54* # DISPLAY VB54 INITIALLY

TC BANKCALL
CADR GOMARK4

TCF KILLAOT # V34 -- DOES GOTOP00H

TCF MARKCHEX # VB33 -- PROCEED, GOT MARKS, COMPUTE LOS

TCF GETDAT # ENTER -- RECYCLE TO V01N71

MARKCHEX CS MARKSTAT # SET BIT12 TO DISCOURAGE MARKRUPT

MASK BIT12
ADS MARKSTAT
MASK LOW9

TS XYMARK # JAM MARK VAC ADR IN XYMARK FOR AVESTAR

CAF ZERO
TS MKDEX # SET MKDEX ZERO FOR LOS VEC CNTRCA MARKSTAT
MASK PRI03 # SEE IF LAST MK PART COMPLETE

TS L

CAF PRI03 # BITS10 AND 11

EXTEND
RXOR LCHANEXTEND
BZF AVESTAR # LAST PAIR COMPLETE -- TO COMPUTE LOS

CNTCHK CCS MARKCNTR # NO PAIR SHOWING -- SEE IF PAIR IN HOLD

TCF +2 # PAIR BURIED -- DECREMENT COUNTER

TCF MKALARM # NO PAIR -- ALARM

TS MARKCNTR # STORE DECREMENTED COUNTER

AVESTAR CAF BIT12 # INITIALIZE MKDEX FOR STAR LOS COUNTER

ADS MKDEX # MKDEX WAS INITIALIZED ZERO IN MARKCHEX

CS MARKCNTR

EXTEND
MP SIX # GET C(L) = -6 MARKCNTRCS XYMARK
AD L # ADD -- MARK VAC ADR SET IN MARKCHEX

INDEX FIXLOC

TS X1 # JAM -- CDU ADR OF X-MARK IN X1

CA FIXLOC # SET PD POINTER TO ZERO

TS PUSHLOC

TC INTPRET

BON	VLOAD*	SURFFLAG	# IF ON SURFACE COMPUTE VEC AT SURFSTAR
		SURFSTAR	
		1,1	# PUT Y-MARK CDUS IN CDUSPOT FOR TRG*NBSM
STOVL		CDUSPOT	
		12D	# LOAD Y-PLANE VECTOR IN NG
CALL			
		TRG*NBSM	# CONVERT IT TO STABLE MEMBER
PUSH	VLOAD*		
		0,1	# PUT X-MARK CDUS IN CDUSPOT FOR TRG*NBSM
STOVL		CDUSPOT	
		18D	# LOAD X-PLANE VECTOR IN NB
CALL			
		TRG*NBSM	# CONVERT IT TO STABLE-MEMBER
VXV	UNIT		# UNIT(XPSM * YPSM)
STADR			
STORE	24D		
AVEIT	SLOAD	PDVL	# N(NUMBER OF VECs) IN 0-1
		MKDEX	
		24D	# LOAD CURRENT VECTOR
VSR3		V/SC	
		0	
STODL		24D	# VEC/N
		0	
DSU		DDV	
		DP1/8	# (N-1)/N
VXSC		VAD	
		STARAD +6	# ADD VEC TO PREVIOUSLY AVERAGED VECTOR
		24D	# (N-1)/N AVESTVEC + VEC/N
STORE		STARAD +6	# AVERAGE STAR VECTOR
STORE		STARSAV2	
EXIT			
CCS	MARKCNTR		# SEE IF ANOTHER MARK PAIR IN MKVAC
TCF	AVESTAR -1		# THERE IS -- GO GET IT -- DECREMENT COUNTER
ENDMARKS	CAF	FIVE	# NO MORE MARKS -- TERMINATE AOTMARK
	INHINT		
	TC	WAITLIST	
	EBANK=	XYMARK	
	2CADR	MKRELEAS	
	TC	ENDMARK	
MKALARM	TC	ALARM	# NOT A PAIR TO PROCESS -- DO GETMKS
	OCT	111	
	TCF	GETMKS	
VO1N71	VN	171	
VO6N87*	VN	687	

MARKRUPT IS ENTERED FROM INTERRUPT LEAD-INS AND PROCESSES CHANNEL 16
CAUSED BY X,Y MARK OR MARK REJECT OR BY THE RATE OF DESCENT SWITCH

MARKRUPT	TS	BANKRUPT	
	CA	CDUY	# STORE CDUS AND TIME NOW -- THEN SEE IF
	TS	ITEMP3	# WE NEED THEM
	CA	CDUZ	
	TS	ITEMP4	
	CA	CDUX	
	TS	ITEMP5	
	EXTEND		
	DCA	TIME2	
	DXCH	ITEMP1	
	XCH	Q	
	TS	QRUPT	
	CAF	OCT34	# SEE IF X OR Y MARK OR MKREJECT
	EXTEND		
	RAND	NAVKEYIN	
	CCS	A	
	TCF	+2	# ITS A LIVE ONE -- SEE IF ITS WANTED
	TCF	SOMEKEY	# ITS SOME OTHER KEY
	CAF	BIT12	# ARE WE ASKING FOR A MARK
	MASK	MARKSTAT	
	CCS	A	
	TC	RESUME	# DON'T WANT MARK OR MKREJECT -- DO NOTHING
	CCS	MARKSTAT	# ARE MARKS BEING ACCEPTED
	TCF	FINDKEY	# THEY ARE -- WHICH ONE IS IT
	TC	ALARM	# MARKS NOT BEING ACCEPTED -- DO ALARM
	OCT	112	
	TC	RESUME	
FINDKEY	CAF	BIT5	# SEE IF MARK REJECT.
	EXTEND		
	RAND	NAVKEYIN	
	CCS	A	
	TCF	MKREJ	# IT'S A MARK REJECT
	CAF	BIT4	# SEE IF Y MARK
	EXTEND		
	RAND	NAVKEYIN	
	CCS	A	
	TCF	YMKRUPT	# IT'S A Y MARK
	CAF	BIT3	# SEE IF X MARK
	EXTEND		
	RAND	NAVKEYIN	

	CCS	A	
	TCF	XMKRUPT	# IT'S A X MARK
SOMEKEY	CAF	OCT140	# NOT MARK OR MKREJECT -- SEE IF DESCENT BITS
	EXTEND		
	RAND	NAVKEYIN	
	EXTEND		
	BZF	+3	# IF NO BITS
	TC	POSTJUMP	# IF DESCENT BITS
	CADR	DESCBITS	
	TC	ALARM	# NO INBITS IN CHANNEL 16.
	OCT	113	
	TC	RESUME	
XMKRUPT	CAF	ZERO	
	TS	RUPTREG1	# SET X MARK STORE INDEX TO ZERO
	CAF	BIT10	
	TCF	+4	
YMKRUPT	CAF	ONE	
	TS	RUPTREG1	# SET Y MARK STORE INDEX TO ONE
	CAF	BIT11	
	TS	XYMARK	# SET MARK IDENTIFICATION
	TC	MARKTYPE	# SEE IF SURFACE MARK
	TCF	SURFSTOR	# SURFACE MARK -- JUST STORE CDUS
	CAF	BIT14	# GOT A MARK -- SEE IF MARK PARI MADE
	MASK	MARKSTAT	
	EXTEND		
	BZF	VERIFYMK	# NOT A PAIR, NORMAL PROCEDURE
	CS	MARKCNTR	# GO A PAIR, SEE IF ANOTHER CAN BE MADE
	AD	FOUR	# IF SO, INCREMENT POINTER, CLEAR BITS 10,11
	EXTEND		
	BZMF	5MKALARM	# HAVE FIVE MARK PAIRS -- DON'T ALLOW MARK
	INCR	MARKCNTR	# OK FOR ANOTHER PAIR, INCR POINTER
	CS	PRI023	# CLEAR BITS 10,11,14 FOR NEXT PAIR
	MASK	MARKSTAT	
	TS	MARKSTAT	
VERIFYMK	CA	XYMARK	
	MASK	MARKSTAT	
	CCS	A	
	TCF	+2	# THIS MARK NOT DESIRED
	TCF	VACSTOR	# MARK DESIRED -- STORE CDUS
	TC	ALARM	
	OCT	114	
	TC	RESUME	# RESUME -- DISPLAY UNCHANGED -- WAIT FOR ACTION



AOTMARK

PAGE 256

1412THE

1					1
2	5MKALARM	TC	ALARM	# ATTEMPTING TO MAKE MORE THAN 5 MK PAIRS	2
3		OCT	107		3
4		TC	MARKTYPE	# SEE IF SURFACE MARK	4
5		TCF	DSPV6N79	# IT IS	5
6		TC	RESUME	# DON'T CHANGE DISPLAY -- DO NOTHING	6
7					7
8					8
9					9
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

AOTMARK

MKREJ	TC TCF	MARKTYPE SURFREJ	# SEE IF SURFACE # SURFACE -- JUST CHECK MARK COUNTER
	CAF MASK	PRI03 MARKSTAT	# INFLIGHT -- SEE IF MARKS MADE
REJALM	CCS TCF TC	A REJECT ALARM	# MARKS MADE -- REJECT ONE # NO MARK TO REJECT -- BAD PROCEDURE -- ALARM
	OCT TC	115 RESUME	# DESIRED ACTION DISPLAYED
REJECT	CS MASK AD	PRI030 MARKSTAT BIT13	# ZERO BIT14, SHOW REJ., SEE IF MARK SINCE # LAST REJECT
	XCH MASK CCS	MARKSTAT BIT13 A	
	TCF	REJECT2	# ANOTHER REJECT SET BIT 10+11 TO ZERO
RENEWMK	CS MASK TS TCF	XYMARK MARKSTAT MARKSTAT REMARK	# MARK MADE SINCE REJECT -- REJECT MARK IN 1D # GO REQUEST NEW MARK ACTION
REJECT2	CS TCF	PRI03 RENEWMK	# ON SECOND REJECT -- DISPLAY VB53 AGAIN
SURFREJ	CCS TCF	MARKCNTR +2	# IF MARK DECREMENT COUNTER
	TCF TS TC	REJALM MARKCNTR RESUME	# NO MARKS TO REJECT -- ALARM

MARKTYPE TESTS TO SEE IF LEM ON LUNAR SURFACE. IF IT IS RETURN TO LOC+1

MARKTYPE CS FLAGWRD8 # SURFFLAG ***** TEMPORARY *****

MASK BIT8

CCS A

INCR

Q

IF SURFACE MARK RETURN TO LOC +1

TC

Q

IF INFLIGHT MARK RETURN TO LOC +2

SURFSTOR

CAF

ZERO

FOR SURFACE MARK ZERO MARK KIND INDEX

TS

RUPTREG1

CS

MARKSTAT

SET BITS10,11 TO SHOW SURFACE MARK

MASK

PRIO3

FOR MARKCHEX

ADS

MARKSTAT

VACSTOR

CAF

LOW9

STORE MARK VAC ADR IN RUPTREG2

MASK

MARKSTAT

TS

RUPTREG2

EXTEND

DCA

ITEMP1

PICK UP MARKTIME

DXCH

TSIGHT

STORE LAST MARK TIME

CA

MARKCNTR

6 X MARKCNTR FOR STORE INDEX

EXTEND

MP

SIX

XCH

L

GET INDEX FROM LOW ORDER PART

AD

RUPTREG2

SET CDU STORE INDEX TO MARKVAC

ADS

RUPTREG1

INCREMENT VAC PICKUP BY MARK FOR FLIGHT

TS

MKDEX

STORE HERE IN CASE OF SURFACE MARK

CA

ITEMP3

INDEX

RUPTREG1

TS

0

STORE CDUY

CA

ITEMP4

INDEX

RUPTREG1

TS

2

STORE CDUZ

CA

ITEMP5

INDEX

RUPTREG1

TS

4

STORE CDUX

TC

MARKTYPE

IF SURFACE MARK -- JUST DO SURFJOB

TCF

SURFJOB

CAF

BIT13

CLEAR BIT13 TO SHOW MARK MADE

AD

XYMARK

SET MARK ID IN MARKSTAT

COM

MASK

MARKSTAT

AD

XYMARK

TS

MARKSTAT

MASK

PRIO3

SEE IF X, Y MARK MADE

TS

L

CA EXTEND	PRI03	
RXOR	LCHAN	
CCS	A	
TCF	REMARK	# NOT PAIR YET, DISPLAY MARK ACTION
CS	MARKSTAT	# MARK PAIR COMPLETE -- SET BIT14
MASK	BIT14	
ADS	MARKSTAT	
TCF	REMARK	# GO DISPLAY V54

AOTMARK

1					1
2	REMARK	CAF	PRI03	# BITS 10 AND 11	2
3		MASK	MARKSTAT		3
4		EXTEND			4
5		MP	BIT6	# SHIFT MARK IDS TO BE 0 TO 3 FOR INDEX	5
6		TS	MKDEX	# STORE VERB INDEX	6
7	SURFJOB	CAF	PRI015		7
8		TC	NOVAC	# ENTER JOB TO CHANGE DISPLAY TO	8
9		EBANK=	XYMARK	# REQUEST NEXT ACTION	9
10		2CADR	CHANGEVB		10
11					11
12		TC	RESUME		12
13					13
14	CHANGEVB	TC	MARKTYPE		14
15		TCF	DSPV6N79	# SURFACE -- DISPLAY V 06 N 79	15
16		INDEX	MKDEX	# INFLIGHT -- PICK UP MARK VB INDEX	16
17		CAF	MKVB54		17
18		TC	PASTIT	# PASTE UP NEXT MK VERB DISPLAY	18
19					19
20	# THE FOUR MKVBS ARE INDEXED -- THEIR ORDER CANNOT BE CHANGED				20
21					21
22	MKVB54	VN	5471	# MAKE X OR Y MARK	22
23	MKVB53	VN	5371	# MAKE Y MARK	23
24	MKVB52	VN	5271	# MAKE X MARK	24
25	MKVB54*	VN	5471	# MAKE X OR Y MARK	25
26	DP1/8	2DEC	.125		26
27					27
28	OCT34	OCT	34		28
29	V06N71	VN	671		29
30	V06N79*	VN	679		30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

ROUTINE TO REQUEST CURSOR AND SPIRAL MEASUREMENTS

COUNT* \$\$/R59

DSPV6N79

CAF

V06N79*

CURSOR -- SPIRAL DISPLAY

TC

BANKCALL

CADR

GOMARKF

TCF

KILLAOT

V34 -- DOES GOTOP00H

TCF

SURFEND

V33 -- PROCEED, END MARKING

CAF

BIT6

IF V32(OCT40) IN MPAC DO RECYCLE

MASK

MPAC

OTHERWISE IT IS LOAD VB ENTER SO

CCS

A

RE-DISPLAY V06N79

TCF

SURFAGAN

VB32 -- RECYCLE

TCF

DSPV6N79

ENTER

SURFEND

CS

BIT14

SET BIT14 TO SHOW MARK END

MASK

MARKSTAT

AD

BIT14

TS

MARKSTAT

SURFAGAN

CA

CURSOR

HOLDS VAC AREA POINTER FOR SURF MARKING

INDEX

MKDEX

TS

1

STORE CURSOR SP 2COMP

CA

SPIRAL

INDEX

MKDEX

TS

3

STORE SPIRAL

CS

MARKSTAT

IF BIT 14 SET -- END MARKING

MASK

BIT14

EXTEND

BZF

MARKCHEX

CA

MARKCNTR

THIS IS RECYCLE -- SEE IF 5 MARKS ALREADY

AD

ONE

COM

AD

FIVE

EXTEND

BZMF

5MKALARM

CAN'T RECYCLE -- TOO MANY MARKS -- ALARM

INCR

MARKCNTR

OF FOR RECYCLE -- INCR COUNTER

TCF

GETMKS +3

GO DISPLAY MARK VB

	BANK	7	
	SETLOC	EXTVERBS	
	BANK		
	EBANK=	OGC	
	COUNT*	\$\$/EXTVB	
# FAN-OUT			
GOEXTVB	INDEX	MPAC	# VERB-40 IS IN MPAC
	TC	LST2FAN	# FAN AS BEFORE.
LST2FAN	TC	VBZERO	# VB40 ZERO (USED WITH NOUN 20 OR 72 ONLY)
	TC	VBCOARK	# VB41 COARSE ALIGN (USED WITH NOUN 20 OR 72 ONLY)
	TC	IMUFINEK	# VB42 FINE ALIGN IMU
	TC	IMUATTCK	# VB43 LOAD IMU ATTITUDE ERROR METERS.
	TC	RRDESEND	# VB44 TERMINATE CONTINUOUS DESIGNATE
	TC	ALM/END	# VB45 SPARE
	TC	ALM/END	# VB46 SPARE
	TC	V47TXACT	# VB47 AGS INITIALIZATION
	TC	DAPDISP	# VB48 LOAD A/P DATA
	TCF	CREWMANU	# VB49 START AUTOMATIC ATTITUDE MANEUVER
	TC	GOLOADLV	# VB50 PLEASE PERFORM
	TC	ALM/END	# VB51 SPARE
	TC	GOLOADLV	# VB52 PLEASE MARK X -- RETICLE.
	TC	GOLOADLV	# VB53 PLEASE MARK Y -- RETICLE.
	TC	GOLOADLV	# VB54 PLEASE MARK X OR Y RETICLE
	TC	ALINTIME	# VB55 ALIGN TIME
	TC	TRMTRACK	# VB56 TERMINATE TRACKING -- P20 + P25
	TC	LRON	# VB57 PERMIT LANDING RADAR UPDATES
	TC	LROFF	# VB58 INHIBIT LANDING RADAR UPDATES
	TC	ALM/END	# VB59 SPARE
	TC	LRPOS2K	# VB60 COMMAND LR TO POSITION 2.
	TC	DAPATTER	# VB61 DISPLAY DAP ATTITUDE ERROR
	TC	TOTATTER	# VB62 DISPLAY TOTAL ATTITUDE ERROR
	TC	R04	# VB63 SAMPLE RADAR ONCE PER SECOND
	TC	VB64	# VB64 CALCULATE, DISPLAY S-BAND ANT ANGLES
	TC	SNUFFOUT	# VB65 DISABLE U,V JETS DURING DPS BURNS.
	TC	ATTACHED	# VB66 ATTACHED MOVE THIS TO OTHER STATE
	TC	V67	# VB67 W MATRIX MONITOR
	TC	ALM/END	# VB68 SPARE
VERB69	TC	VERB69	# VB69 FORCE A HARDWARE RESTART
	TC	V70UPDAT	# VB70 UPDATE LIFTOFF TIME.
	TC	V71UPDAT	# VB71 UNIVERSAL UPDATE -- BLOCK ADDRESS.
	TC	V72UPDAT	# VB72 UNIVERSAL UPDATE -- SINGLE ADDRESS.
	TC	V73UPDAT	# VB73 UPDATE AGC TIME (OCTAL).
	TC	DNEDUMP	# VB74 INITIALIZE DOWN-TELEMETRY PROGRAM FOR ERASABLE DUMP.
	TC	OUTSNUFF	# VB75 ENABLE U,V JETS DURING DPS BURNS.

TC	MINIMP	# VB76 MINIMUM IMPULSE MODE
TC	NOMINIMP	# VB77 RATE COMMAND MODE
TC	R77	# VB78 START LR SPURIOUS RETURN TEST
TC	R77END	# VB79 TERMINATE LR SPURIOUS RETURN TEST
TC	LEMVEC	# VB80 UPDATE LEM STATE VECTOR
TC	CSMVEC	# VB81 UPDATE CSM STATE VECTOR
TC	V82PERF	# VB82 REQUEST ORBIT PARAM DISPLAY (R30)
TC	V83PERF	# VB83 REQUEST REND PARAM DISPLAY (R31)
TC	ALM/END	# VB84 SPARE
TC	VERB85	# VB85 DISPLAY RR LOS AZ AND ELEV
TC	ALM/END	# VB86 SPARE
TC	ALM/END	# VB87 SPARE
TC	ALM/END	# VB88 SPARE
TC	V89PERF	# VB89 ALIGN XORZ LEM AXIS ALONG LOS (R63)
TC	V90PERF	# VB90 OUT OF PLANE RENDEZVOUS DISPLAY
TC	GOSHOSUM	# VB91 DISPLAY BANK SUM.
TC	SYSTEST	# VB92 OPERAT IMU PERFORMANCE TEST.
TC	WMATRXNG	# VB93 CLEAR RENDWFLG
TC	ALM/END	# VB94 SPARE
TC	UPDATOFF	# VB95 NO STATE VECTOR UPDATE ALLOWED
TC	VERB96	# VB96 INTERRUPT INTEGRATION AND GO TO P00
TC	GOLOADLV	# VB97 PLEASE VERIFY ENGINE FAILURE
TC	ALM/END	# VB98 SPARE
TC	GOLOADLV	# VB99 PLEASE ENABLE ENGINE

END OF EXTENDED VERB FAN

TESTXACT	CCS	EXTVBACT	# ARE EXTENDED VERBS BUSY
	TC	ALM/END	# YES, TURN ON OPERATOR LIGHT
	CA	FLAGWRD4	# ARE PRIORITY DISPLAYS USING DSKY
	MASK	OC24100	
	CCS	A	
	TC	ALM/END	# YES
SETXTACT	CAF	OCT24	# SET 3, AND 5
	TS	EXTVBACT	# NO. SET FLAG TO SHOW EXT VERB DISPLAY
			# SYSTEM BUSY
	CA	Q	
	TS	MPAC +1	
	CS	TWO	# BLANK EVERYTHING EXCEPT MM AND VERB
	TC	NVSUB	
	TC	+1	
	TC	MPAC +1	
XACTALM	TC	FALTON	# TURN ON OPERATOR ERROR LIGHT.
	TC	ENDEXT	# RELEASE MARK AND EXT. VERB DISPLAY SYS.
TERMEXTV	EQUALS	ENDEXT	

EXTENDED_VERBS

1					1
2	ENDEXTVB	EQUALS	ENDEXT		2
3					3
4	XACTO	CAF	ZERO	# RELEASE MARK AND EXT. VERB DISPLAY SYS.	4
5		TC	SETXTACT		5
6					6
7	ALM/END	TC	FALTON	# TURN ON OPERATOR ERROR LIGHT	7
8	GOPIN	TC	POSTJUMP		8
9		CADR	PINBRNCH		9
10					10
11	CHKPOOH	CA	MODREG	# CHECK FOR P00 OR P00-.	11
12		EXTEND			12
13		BZF	TCQ		13
14		TC	ALM/END		14
15					15
16	OC24100	OCT	24100		16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

#	VBZERO	VERB 40	DESCRIPTION
#			
#	1.	REQUIRE NOUN 20 (ICDU ANGLES) OR NOUN 72 (RCDU ANGLES).	
#	2.	FOR N20, CHECK IMUCADR IN AN EFFORT TO AVOID A 1210 RESTART.	
#		FOR N72, CHECK IF EITHER RADAR IS IN USE.	
#	3.	EXECUTE THE CDU ZERO.	
#	4.	STALL UNTIL THE ZERO IS DONE.	
#	5.	DON'T DIFFERENTIATE BETWEEN A BAD OR GOOD RETURN.	
#	6.	EXIT, RE-ESTABLISHING THE INTERRUPTED DISPLAY (IF ANY).	

VBZERO	TC	OP/INERT	
	TC	IMUZEROK	# RETURN HERE IF NOUN = ICDU(20)
	TC	RRZEROK	# RETURN HERE IF NOUN = RCDU(72)
IMUZEROK	TC	CKMODCAD	
	TC	BANKCALL	# KEYBOARD REQ FOR ISS CDUZERO
	CADR	IMUZERO	
	TC	BANKCALL	# STALL
	CADR	IMUSTALL	
	TC	+1	
	TC	GOPIN	# IMUZERO
RRZEROK	TC	RDRUSECK	
	TC	BANKCALL	
	CADR	RRZERO	
RWAITK	TC	BANKCALL	
	CADR	RADSTALL	
	TCF	+1	
	TC	GOPIN	# RRZERO

#	LRPOS2K	VERB 60	DESCRIPTION
#	COMMAND	LANDING RADAR TO POSITION 2	
#			
#	1.	EXIT WITH OP ERROR IF SOMEONE IS USING EITHER RADAR.	
#	2.	ALARM WITH CODE 523 IF POS 2 IS NOT INDICATED WITHIN	
#		THE PRESCRIBED TIME.	
#	3.	RE-ESTABLISH THE DISPLAYS.	
LRPOS2K	TC	RDRUSECK	
	TC	BANKCALL	# COMMAND LR TO POSITION 2
	CADR	LRPOS2	
	TC	BANKCALL	
	CADR	RADSTALL	
	TC	LRP2ALM	
	TC	GOPIN	
LRP2ALM	TC	ALARM	
	OCT	523	
	TC	GOPIN	



1				1
2				2
3	# V61 VERB 61, DISPLAY DAP ATTITUDE ERRORS ON FDAI ATTITUDE ERROR NEEDLES.			3
4				4
5	DAPATTER	TC	DOWNFLAG	5
6		ADRES	NEEDLFLG	6
7		TC	GOPIN	7
8				8
9	# V62 VERB 62, DISPLAY TOTAL ATTITUDE ERRORS ON FDAI ATTITUDE ERROR NEEDLES.			9
10				10
11	TOTATTER	TC	UPFLAG	11
12		ADRES	NEEDLFLG	12
13		TC	GOPIN	13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

```
1  # VBCOARK      VERB 41      DESCRIPTION
2  #      COARSE ALIGN IMU OR RADAR
3
4  #
5  #      1.      REQUIRE NOUN 20 OR NOUN 72 OR TURN ON OPERATOR ERROR.
6  #      2.      REQUIRE EXT VERB DISPLAY SYS AVAILABLE OR TURN ON OPERATOR ERROR LIGHT AND GO TO PINBRNCH.
7
8  #      CASE 1, NOUN 20 (ICDU ANGLES)
9  #      3.      SET EXT VERB DISPLAY ACTIVE FLAG.
10 #      4.      DISPLAY FLASHING V25,N22 (LOAD NEW ICDU ANGLES).
11
12 #      RESPONSES
13 #      A.      TERMINATE
14 #      1.      RELEASE EXT VERB DISPLAY SYSTEM
15
16 #      B.      PROCEED
17 #      1.      COARSE ALIGN TO THE EXISTING THETAD'S (ICORK2).
18
19 #      C.      ENTER
20 #      1.      COARSE ALIGN TO THE LOADED THETAD'S (ICORK2).
21
22 # ICORK2
23 #      1.      RE-DISPLAY VERB 41.
24 #      2.      EXECUTE IMUCCARS (IMU COARSE ALIGN).
25 #      3.      EXECUTE IMUSTALL (ALLOW TIME FOR DATA TRANSFER).
26 #      4.      RELEASE EXT VERB DISPLAY SYSTEM.
27
28 #      CASE 2, NOUN 72 (RCDU ANGLES)
29 #      5.      EXIT WITH OP ERROR IF SOMEONE IS USING EITHER RADAD.
30 #      DISPLAY FLASHING V24, N73 (LOAD NEW RR TRUNION ANGLE AND NEW SHAFT ANGLE).
31
32 #      RESPONSES
33 #      A.      TERMINATE
34 #      1.      RELEASE EXT VERB DISPLAY SYS.
35
36 #      B.      PROCEED OR ENTER
37 #      1.      EXECUTE AURLOKON (ASK OPERATOR FOR LOCK-ON REQUIREMENTS).
38 #      2.      RE-DISPLAY VERB 41.
39 #      3.      SCHEDULE RRDESK2 WITH PRIORITY 20.
40 #      4.      RELEASE EXT VERB DISPLAY SYS.
41
42 # AURLOKON
43 #      1.      FLASH V04 N12 R1 = 00006 R2 = 00002
44 #      RESPONSES
45 #      A.      TERMINATE
46 #      B.      PROCEED
47 #      1.      RESET LOCK-ON SWITCH
48 #      2.      SET CONTINUOUS DESIGNATE FLAG
49 #      3.      DISABLE R25
50 #      C.      V22 E 1 E, R1 = 00001, PROCEED
51 #      1.      SET LOCK-ON SWITCH
52
53 VBCOARK      TC      OP/INERT
54 TC      IMUCOARK      # RETURN HERE IF NOUN = ICDU (20)
55 TC      RRDESNBK      # RETURN HERE IF NOUN = RCDU (72)
56
57 # RETURNS TO L+1 IF IMU OR L+2 IF RR.
58
59 OP/INERT      CS      OCT24
60 AD      NOUNREG
61
62 EXTEND
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

[illegible]

DESIGNATE TO DESIRED GIMBAL ANGLES.

RRDESNBK	TC	RDRUSECK	
	TC	TESTXACT	
	CA	RNDVZBIT	# IS P20 RUNNING?
	MASK	FLAGWRDO	
	CCS	A	
	TCF	XACTALM	# OPERADOR ERROR IF IN P20
	CS	OCT41000	# TERMINATE PRESENT DESIGNATION
	INHINT		# RELINT DONE IN GOXDSPF
	MASK	RADMODES	
	TS	RADMODES	
	CAF	VNLDRCDU	# ASK FOR GIMBAL ANGLES.
	TC	BANKCALL	
	CADR	GOXDSPF	
	TC	TERMEXTV	
	TCF	-4	# V33
	TC	BANKCALL	# ASK OP FOR LOCK ON REQUIREMENTS.
	CADR	AURLOKON	
	CAF	OPTCOARV	# RE-DISPLAY OUR OWN VERB
	TC	BANKCALL	
	CADR	EXDSPRET	
	CAF	PRI020	
	TC	FINDVAC	
	EBANK=	LOSCOUNT	
	2CADR	RRDESK2	
	TCF	TERMEXTV	# FREES DISPLAY
VNLDRCDU	VN	2473	
OPTCOARV	EQUALS	IMUCOARV	# DIFFERENT NOUNS.
RRDESK2	TC	BANKCALL	
	CADR	RRDESNB	
	TC	+1	# DUMMY NEEDED SINCE DESRETRN DOES INCR
	CA	PRIORITY	
	MASK	LOW9	
	CCS	A	
	INDEX	A	
	TS	A	# RELEASE THIS JOB'S VAC AREA.
	COM		# INSURE ENDOFJOB DOES A NOVAC END (BZMF).
	ADS	PRIORITY	
	TC	BANKCALL	# WAIT FOR COMPLETION OF DESIGNATE
	CADR	RADSTALL	

1					1
2		TC	+2	# BADEND -- NO LOCKON OR OUT OF LIMITS	2
3		TC	ENDOFJOB	# GOODEND -- LOCKON ACHIEVED	3
4		TC	ALARM		4
5		OCT	503	# TURN ON ALARM LIGHT -- 503 DESIGNATE FAIL	5
6					6
7		TC	ENDOFJOB		7
8					8
9	RRDESEND	CCS	RADMODES	# TERMINATE CONTINOUS DESIGNATE ONLY	9
10		TCF	GOPIN		10
11		TCF	GOPIN		11
12		TCF	+1		12
13		CS	OCT41000	# BEGDES GOES TO ENDRADAR	13
14		INHINT		# RELINT DONE IN DOWNFLAG	14
15		MASK	RADMODES		15
16		TS	RADMODES		16
17		TC	CLRADMOD		17
18		CAF	1SEC		18
19		TC	BANKCALL		19
20		CADR	DELAYJOB		20
21		TC	DOWNFLAG	# ENABLE R25 GIMBAL MONITOR	21
22		ADRES	NORRMON		22
23		TCF	GOPIN		23
24	OCT41000	OCT	41000	# CONTINOUS DESIGNATE -- DESIGNATE	24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

	BANK	23	
	SETLOC	EXTVB1	
	BANK		
	COUNT*	\$\$/EXTVB	
AURLOKON	TC	MAKECADR	
	TS	DESRET	
	CAF	TWO	
	TS	OPTIONX +1	
	CAF	SIX	# OPTION CODE FOR V04N12
	TS	OPTIONX	
-5	CAF	V04N1272	
	TC	BANKCALL	# R2 00001 LOCK-ON
	CADR	GOMARKFR	
	TCF	ENDEXT	# V34
	TCF	+5	# V33
	TCF	-5	# V32
	CAF	BIT3	
	TC	BLANKET	
	TC	ENDOFJOB	
+5	CA	OPTIONX +1	
	MASK	BIT2	
	CCS	A	
	TCF	NOLOKON	
	TC	UPFLAG	
	ADRES	LOKONSW	
	TCF	AURLKON1	
NOLOKON	TC	DOWNFLAG	# IF NO LOCK-ON, SET BIT15 OF RADMADES TO
	ADRES	LOKONSW	# INDICATE THAT CONTINUOUS DESIGNATION IS
	TC	UPFLAG	# WANTED (TO BE TERMINATED BY V44.)
	ADRES	CDESFLAG	
	TC	UPFLAG	# SET NO RR ANGLE MONITOR FLAG.
	ADRES	NORRMON	# DISABLE R25 RR GIMBAL MONITOR IN T4RUPT
AURLKON1	RELINT		
	CA	DESRET	
	TCF	BANKJUMP	
V04N1272	VN	412	
-LOKONFG	OCT	-20	
	BANK	43	
	SETLOC	EXTVERBS	
	BANK		
	COUNT*	\$\$/EXTVB	
LRON	TC	UPFLAG	# PERMIT INCORPORATION OF LR DATA V57

1412THE

#	IMUFINEK	VERB 42	DESCRIPTION
#	FINE ALIGN IMU		
#			
#	1.	REQUIRE EXT VERB DISPLAY AVAILABLE AND SET BUSY FLAG OR TURN ON OPER ERROR AND GO TO PINBRNCH.	
#	2.	DISPLAY FLASHING V25,N93....LOAD DELTA GYRO ANGLES....	
#	RESPONSES		
#	A.	TERMINATE	
#	1. RELEASE EXT VERB DISPLAY SYSTEM.		
#	B.	PROCEED OR ENTER	
#	1. RE-DISPLAY VERB 42		
#	2. EXECUTE IMUFINE (IMU FIVE ALIGN MODE SWITCHING).		
#	3. EXECUTE IMUSTALL (ALLOW FOR DATA TRANSFER)		
#	A.	FAILED	
#	1. RELEASE EXT VERB DISPLAY SYSTEM.		
#	B.	GOOD	
#	1. EXECUTE IMUPULSE (TORQUE IRIGS).		
#	2. EXECUTE IMUSTALL AND RELEASE EXT VERB DISPLAY SYSTEM.		
IMUFINEK	TC	CKMODCAD	
	TC	TESTXACT	# FINE ALIGN WITH GYRO TORQUING.
	CAF	VNLODGYR	# CALL FOR LOAD OF GYRO COMMANDS
	TC	BANKCALL	
	CADR	GOXDSPF	
	TC	TERMEXTV	
	TC	+1	# PROCEED WITHOUT A LOAD
	CAF	IMUFINEV	# RE-DISPLAY OUR OWN VERB
	TC	BANKCALL	
	CADR	EXDSPRET	
	TC	BANKCALL	# CALL MODE SWITCH PROG
	CADR	IMUFINE	
	TC	BANKCALL	# HIBERNATION
	CADR	IMUSTALL	
	TC	ENDEXTVB	
FINEK2	CAF	LGYROBIN	# PINBALL LEFT COMMANDS IN OGC REGISTERS
	TC	BANKCALL	
	CADR	IMUPULSE	
	TC	BANKCALL	# WAIT FOR PULSES TO GET OUT.
	CADR	IMUSTALL	
	TC	ENDEXTVB	
	TC	ENDEXTVB	
LGYROBIN	ECADR	OGC	
VNLODGYR	VN	2593	
IMUFINEV	VN	4200	

#	GOLOADLV	VERB 50	DESCRIPTION
#	AND OTHER PLEASE		

EXTENDED_VERBS

DO SOMETHING VERBS

PLEASE PERFORM, MARK, CALIBRATE, ETC.

1. PRESSING ENTER ON DSKY INDICATES REQUESTED ACTION HAS BEEN PERFORMED, AND THE PROGRAM DOES THE SAME RECALL AS A COMPLETED LOAD.

2. THE EXECUTION OF A VERB 33 (PROCEED WITHOUT DATA) INDICATES THE REQUESTED ACTION IS NOT DESIRED.

SBANK= PINSUPER # FOR LOADLV1 AND SHOWSUM CADR'S

GOLOADLV TC FLASHOFF

CAF PINSUPBT
EXTEND

WRITE SUPERBNK
TC POSTJUMP
CADR LOADLV1

VERB 47 -- AGS INITIALIZATION -- R47.

SEE LOG SECTION AGS INITIALIZATION FOR OTHER PERTINENT REMARKS.

V47TXACT TC TESTXACT # NO OTHER EXTVERB.

CAF PRI04
TC FINDVAC
SBANK= LOWSUPER
EBANK= AGSBUFF
2CADR AGSINIT

TC ENDOFJOB

CKMODCAD CA MODECADR

EXTEND
BZF TCQ
TC ALM/END # SOMEBODY IS USING MODECADR SO EXIT

#	ALINTIME	VERB 55	DESCRIPTION
#	1.	SET EXT VERB DISPLAY BUSY FLAG.	
#	2.	DISPLAY FLASHING V25,N24 (LOAD DELTA TIME FOR AGC CLOCK.	
#	3.	REQUIRE EXECUTION OF VERB 23.	
#	4.	ADD DELTA TIME, RECEIVED FROM INPUT REGISTER, TO THE COMPUTER TIME.	
#	5.	RELEASE EXT VERB DISPLAY SYSTEM	
ALINTIME	TC	TESTXACT	
	TC	POSTJUMP	# NO ROOM IN 43
	CADR	R33	
	BANK	42	
	SETLOC	SBAND	
	BANK		
	COUNT*	\$\$/R33	
R33	CAF	PRI07	
	TC	PRI0CHNG	
	CAF	VNLODDT	
	TC	BANKCALL	
	CADR	GOXDSPF	
	TC	ENDEXT	# TERMINATE
	TC	ENDEXT	# PROCEED
	CS	DEC23	# DATA IN OR RESEQUENCE (UNLIKELY)
	AD	MPAC	# RECALL LEFT VERB IN MPAC
	EXTEND		
	BZF	UPDATIME	# GO AHEAD WITH UPDATE ONLY IF RECALL
	TC	ENDEXT	# WITH V23 (DATA IN).
UPDATIME	INHINT		# DELTA TIME IS IN DSPTM1, +1.
	CAF	ZERO	
	TS	MPAC +2	# NEEDED FOR TP AGREE
	TS	L	# ZERO T1 + 2 WHILE ALIGNING.
	DXCH	TIME2	
	DXCH	MPAC	
	DXCH	DSPTM2 +1	# INCREMENT
	DAS	MPAC	
	TC	TPAGREE	# FORCE SIGN AGREEMENT.
	DXCH	MPAC	# NEW CLOCK.
	DAS	TIME2	
	RELINT		
UPDTMEND	TC	ENDEXT	
DEC23	DEC	23	# V 23
VNLODDT	VN	2524	# V25N24 FOR LOAD DELTA TIME

SET UP FOR RADAR SAMPLING.

BANK 42
SETLOC EXTVERBS
BANK

EBANK= RSTACK

COUNT* \$\$/R0477

R77 TC RDRUSECK # TRY TO AVOID THE 1210.

CA FLAGWRD3 # IS R04 RUNNING?
MASK R04FLBIT
CCS ATC ALM/END # YES.
TC UPFLAG
ADRES R77FLAG

TCF R04Z

R04 TC RDRUSECK # TRY TO AVOID THE 1210.

TC TESTXACT
TC UPFLAG
ADRES R04FLAG # SET R04FLAG FOR ALARMS

R04Z

CAF EBANK4
TS EBANKCAF 1SEC+1 # SAMPLE ONCE PER SECOND
TS RSAMPDT
CAF ZEROTS RTSTLOC
TS RFAILCNT # ZERO BAD SAMPLE COUNTERINHINT
CS LRPOSCAL # INITIALIZE
MASK RADMODES # BIT 9 LR RANGE LOW SCALE =0
TS RADMODES # BIT 6 LR POS 1 =0
CAF LRPOSCAL # BIT 3 RR RANGE LOW SCALE =0
EXTENDRAND CHAN33
ADS RADMODES
RELINTCS FLAGWRD3 # CHECK R04FLAG R04 =1 R77 =0
MASK R04FLBITCCS A
TCF R04K

R04A CAF ONE # INDICATES RENDEZVOUS DESIRED

TS OPTIONX +1
CAF BIT3 # OPTION CODE FOR V04N12

1	# EXTENDED_VENDS										1
2		TS	OPTIONX								2
3		CAF	V04N12X								3
4		TC	BANKCALL	#	R2	00001	RENDEZVOUS RADAR				5
5		CADR	GOMARKFR	#		00002	LANDING RADAR				6
6		TCF	R04END	#	V34						7
7		TCF	+5	#	V33						8
8		TCF	R04A	+2	#	R2					10
9		CAF	BIT3								11
10		TC	BLANKET								13
11		TC	ENDOFJOB								14
12											15
13		CA	OPTIONX +1	#	SAVE DESIRED OPTION	RR =1	LR =2				17
14		TS	RTSTDEX								18
15											19
16	R04X	CAF	SIX	#	RR OR LR DESIRED						21
17		MASK	RTSTDEX								22
18		CCS	A								23
19		TCF	R04L	#	LANDING RADAR						25
20		TS	RTSTBASE	#	FOR RR	BASE =0,	MAX =1				26
21											27
22	R04B	CAF	BIT2	#	IS RR AUTO MODE DISCRETE PRESENT						29
23		EXTEND									30
24		RAND	CHAN33								31
25		EXTEND									32
26		BZF	R04C	#	YES						33
27											34
28		CAF	201R04	#	REQUEST SELECTION OF RR AUTO MODE						37
29		TS	DSPTM1								38
30		CAF	V50N25X								39
31		TC	BANKCALL								40
32		CADR	GOMARK4								41
33		TCF	R04END	#	V34						42
34		TCF	R04B	#	V33						43
35		TCF	-7	#	E						44
36											45
37	R04C	CAF	BIT14	#	ENABLE RR AUTO TRACKER						46
38		EXTEND									47
39		WOR	CHAN12								48
40											49
41		CAF	TWO								50
42		TS	RTSTMAX	#	FOR SEQUENTIAL STORAGE						51
43											52
44		TC	WAITLIST								53
45		SBANK=	PINSUPER								54
46		EBANK=	RSTACK								55
47		2CADR	RADSAMP								56
48											57
49		RELINT									58
50											59
51		CS	FLAGWRD3	#	CHECK R04FLAG	R04 =1	R77 =0				60
52		MASK	R04FLBIT								61
53											62
54											63
55											64
56											65
57											66
58											67
59											68
60											69

	CCS	A	
	TCF	GOPIN	# R77
	CAF	SIX	# RR OR LR
	MASK	RTSTDEX	
	CCS	A	
	TCF	R04LR	# LR
R04RR	CAF	V16N72	# DISPLAY RR CDU ANGLES (1/SEC)
	TC	BANKCALL	# R1 +- XXX.XX DEG TRUNNION
	CADR	GOMARKF	# R2 +- XXX.XX DEG SHAFT
	TCF	R04END	# V34 R3 BLANK
	TCF	+2	# V33
	TCF	R04RR	# V32
	CAF	V16N78	# DISPLAY RR RANGE AND RANGE RATE (1/SEC)
	TC	BANKCALL	# R1 +- XXX.XX NM RANGE
	CADR	GOMARKF	# R2 +- XXXXX. FPS RANGE RATE
	TCF	R04END	# V34 R3 BLANK
	TCF	R04Y	# V33
	TCF	R04RR	# V32
R04LR	CAF	V16N66	# DISPLAY LR RANGE AND POSITON (1/SEC)
	TC	BANKCALL	# R1 +- XXXXX, FT LR RANGE
	CADR	GOMARKF	# R2 + 0000X. POS. NO.
	TCF	R04END	# V34 R3 BLANK
	TCF	+2	# V33
	TCF	R04LR	# V32
	CAF	V16N67	# DISPLAY LR VELX, VELY, VELZ (1/SEC)
	TC	BANKCALL	# R1 +- XXXXX. FPS LR V(X)
	CADR	GOMARKF	# R2 +- XXXXX. FPS LR V(Y)
	TCF	R04END	# V34 R3 +- XXXXX. FPS LR V(Z)
	TCF	R04Y	# V33
	TCF	R04LR	# V32
R04Y	CAF	ZERO	# TO TERMINATE SAMPLING.
	TS	RSAMPDT	
	CAF	2SECS	# WAIT FOR LAST RADARUP
	TC	BANKCALL	
	CADR	DELAYJOB	
	CAF	1SEC+1	# SAMPLE ONCE PER SECOND
	TS	RSAMPDT	
	CAF	ZERO	# FOR STORING RESULTS
	TS	RTSTLOC	
	CAF	SIX	
	MASK	RTSTDEX	
	CCS	A	
	CS	ONE	# WAS LR
	AD	TWO	# WAS RR

	TCF	R04X -1	
R04K	CAF TS	250MS+1 RSAMPDT	# SAMPLE 4 LR COMPONENTS PER SECOND.
R04L	CAF TS	TWO RTSTBASE	# FOR LR BASE =2, MAX =3
	CAF TCF	SIX R04C +4	
R04END	CAF TS	ZERO RSAMPDT	# ZERO RSAMPDT # TO TERMINATE SAMPLING
	CAF TC	BIT8 BANKCALL	# WAIT 1.28 SECONDS FOR POSSIBLE # PENDING RUPT.
	CADR	DELAYJOB	
	INHINT		
	CS EXTEND WAND	BIT14 CHAN12	# DISABLE RR AUTO TRACKER.
	TC ADRES	DOWNFLAG R04FLAG	# SIGNAL END OF R04.
	TC	ENDEXT	
R77END	CAF TS CAF	EBANK4 EBANK ZERO	# TO TERMINATE SAMPLING
	TS CAF TC	RSAMPDT BIT6 BANKCALL	# WAIT 320 MS FOR POSSIBLE # PENDING RUPT.
	CADR	DELAYJOB	
	TC ADRES TCF	DOWNFLAG R77FLAG GOPIN	
V16N72	VN	1672	
V16N78	VN	1678	
V16N66	VN	1666	
V16N67	VN	1667	
V04N12X	VN	412	
V50N25X	VN	5025	
201R04	OCT	00201	
1SEC+1	DEC	101	
250MS+1	EQUALS	CALLCODE	
LRPOSCAL	OCT	444	

RDRUSECK	CS MASK	FLAGWRD3 NR29FBIT	# IS R29 ON?
	CCS TC CA	A ALM/END FLAGWRD5	# YES # IS R77 RUNNING?
	MASK CCS TC	R77FLBIT A ALM/END	# YES.
	CS MASK CCS	FLAGWRD7 V37FLBIT A	# IS SERVICER RUNNING AND HENCE POSSIBLY # R12 USING THE LR?
	TCF CS MASK	CHECKRR FLGWRD11 LRBYBIT	# NO # YES, IS R12 ON?
CHECKRR	CCS TC CS	A ALM/END FLAGWRD1	# YES # IS THE TRACK FLAG SET AND HENCE POSSIBLY
	MASK CCS TCF	TRACKBIT A CHECKP22	# P20 USING THE RR? # NO, CHECK FOR P22.
CKRNDBIT	CA MASK	FLAGWRD0 RNDVZBIT	# YES, BUT IS IT P25?
CHECKP22	CCS TC CS	A ALM/END MODREG	
	AD EXTEND BZF	DEC22 ALM/END	
DEC22	TC DEC	Q 22	
	COUNT*	\$\$/EXTVB	
VB64	TC TC CAF	CHKPOOH TESTXACT PRIO4	# DEMAND PROGRAM 00. # IF DISPLAY SYS. NOT BUSY MAKE IT BUSY.
	TC EBANK= 2CADR	FINDVAC ALPHASB SBANDANT	# CALC., DISPLAY S-BAND ANTENNA ANGLES.
	TC	ENDOFJOB	

#	IMUATTCK	VERB 43	DESCRIPTION
#	LOAD IMU ATTITUDE ERROR METERS		
#			
#	1.	REQUIRE P00 OR FRESH START.	
#	2.	REQUIRE COARSE ALIGN ENABLE AND ZERO ICDU BITS OFF.	
#	3.	REQUIRE THAT NEEDLES BE OFF.	
#	4.	REQUEST LOAD OF N22 (VALUES TO BE DISPLAYED).	
#	5.	ON PROCEED OR ENTER RE-DISPLAY V43 AND SEND PULSES.	
IMUATTCK	TC	CHKPOOH	# VB 76 -- LOAD IMU ATT. ERROR METERS
	CAF	BITS4&5	# SEE IF COARSE ALIGN ENABLE AND ZERO IMU
	EXTEND		# CDUS BITS ARE ON
	RAND	CHAN12	
	CCS	A	
	TCF	ALM/END	# NOT ALLOWED IF IMU COARSE OR IMU ZERO ON
	CAF	BIT13-14	# BOTH BITS 13 AND 14 MUST BE 1
	EXTEND		# INDICATING THE MODE SELECTED IS OFF.
	RXOR	CHAN31	
	MASK	BIT13-14	
	EXTEND		
	BZF	+2	# NEEDLES IS OFF.
	TCF	ALM/END	# EXIT. NEEDLES IS ON.
	TC	TESTXACT	
	CAF	VNLODCDU	
	TC	BANKCALL	
	CADR	GOXDSPF	
	TC	ENDEXT	# V34
	TC	+1	
	CAF	V43K	# REDISPLAY OUR VERB.
	TC	BANKCALL	
	CADR	EXDSPRET	
	CAF	BIT6	
	EXTEND		
	WOR	CHAN12	# ENABLE ERROR COUNTERS.
	CAF	TWO	
	TC	WAITLIST	# PUT OUT COMMANDS IN .32 SECONDS.
	EBANK=	THETAD	
	2CADR	ATTCK2	
	TCF	ENDEXT	
	BANK	42	
	SETLOC	PINBALL3	# SOMETHING IN B42.
	BANK		
	COUNT*	\$\$/EXTVB	

ATTCK2 CAF TWO # PUT OUT COMMANDS.
+1 TS Q # CDU WILL LIMIT EXCESS DATA.

INDEX A
CA THETAD
EXTEND

MP ATTSCALE
INDEX Q
XCH CDUXCMD

CCS Q
TCF ATTCK2 +1

CAF 13,14,15
EXTEND
WOR CHAN14

TCF TASKOVER # LEAVE ERROR COUNTERS ENABLED.

ATTSCALE DEC 0.1

BANK 7
SETLOC EXTVERBS
BANK

COUNT* \$\$/EXTVB

V43K VN 4300

V82PERF VERB82 DESCRIPTION
REQUEST ORBIT PARAMETERS DISPLAY (R30)
#

1. IF AVERAGE G IS OFF:
FLASH DISPLAY V04N06. R2 INDICATES WHICH SHIP'S STATE VECTOR IS
TO BE UPDATED. INITIAL CHOICE IS THIS SHIP (R2=1). ASTRONAUT
CAN CHANGE TO OTHER SHIP BY V22EXE, WHERE X NOT EQ I.
SELECTED STATE VECTOR UPDATED BY THISPREC (OTHPREC).
CALLS SR30.1 (WHICH CALLS TFFCONMU + TFFRP/RA) TO CALCULATE
RPER (PERIGEE RADIUS), RAPO (APOGEE RADIUS), HPER (PERIGEE
HEIGHT ABOVE LAUNCH PAD OR LUNAR LANDING SITE), HAPO (APOGEE
HEIGHT AS ABOVE), TPER (TIME TO PERIGEE), TFF (TIME TO
INTERSECT 300 KFT ABOVE PAD OR 35KFT ABOVE LANDING SITE).
FLASH MONITOR V16N44 (HAPO, HPER, TFF). TFF IS -59M59S IF IT WAS
NOT COMPUTABLE, OTHERWISE IT INCREMENTS ONCE PER SECOND.
ASTRONAUT HAS OPTION TO MONITOR TPER BY KEYING IN N 32 E.
DISPLAY IS IN HMS, IS NEGATIVE (AS WAS TFF), AND INCREMENTS
ONCE PER SECOND ONLY IF TFF DISPLAY WAS -59M59S.

2. IF AVERAGE G IS ON:
CALLS SR30.1 APPROX EVERY TWO SECS. STATE VECTOR IS ALWAYS
FOR THIS VEHICLE. V82 DOES NOT DISTURB STATE VECTOR. RESULTS
OF SR30.1 ARE RAPO, RPER, HAPO, HPER, TPER, TFF.
FLASH MONITOR V16N44 (HAPO, HPER, TFF).
IF MODE IS P11, THEN CALL DELRSPL SO ASTRONAUT CAN MONITOR
RESULTS BY N50E. SPLASH COMPUTATION DONE ONCE PER TWO SECS.

V82PERF	TC	TESTXACT	
	CAF	PRI07	# LESS THAN LAMBERT. R30,V82
	TC	PRI0CHNG	
	EXTEND		
	DCA	V82CON	
	TC	SUPDXCHZ	# V82CALL IN DIFF SUPERBANK FROM V82PERF
V82CON	EBANK=	HAP0	
	2CADR	V82CALL	
# VB83PERF	VERB 83	DESCRIPTION	
#	REQUEST	RENDEZVOUS	PARAMETER DISPLAY (R31)
#			
#	1.	SET EXT	VERB DISPLAY BUSY FLAG.
#	2.	SCHEDULE	R31CALL WITH PRIORITY 5.
#	A.	DISPLAY	
#		R1	RANGE
#		R2	RANGE RATE
#		R3	THETA
V83PERF	TC	TESTXACT	
	CAF	BIT2	
	TC	WAITLIST	
	EBANK=	TSTRT	
	2CADR	R31CALL	
	TC	ENDOFJOB	
# VERB 89	DESCRIPTION	RENDEZVOUS	FINAL ATTITUDE ROUTINE (R63)
#			
#	CALLED BY VERB 89 ENTER DURING P00. PRIO 10 IS USED. CALCULATES AND		
#	DISPLAYS FINAL FDAI BALL ANGLES TO POINT LM +X OR +Z AXIS AT CSM.		
#			
#	1. KEY IN V 89 E ONLY IF IN PROG 00. IF NOT IN P00, OPERATOR ERROR AND		
#	EXIT R63, OTHERWISE CONTINUE.		
#			
#	2. IF IN P00, DO IMU STATUS CHECK ROUTINE (R02BOTH). IF IMU ON AND ITS		
#	ORIENTATION KNOWN TO LGC,CONTINUE.		
#			
#	3. FLASH DISPLAY V 04 N 06. R2 INDICATES WHICH SPACECRAFT AXIS IS TO		
#	BE POINTED AT CSM. INITIAL CHOICE IS PREFERRED (+Z) AXIS (R2=1).		
#	ASTRONAUT CAN CHANGE TO (+X) AXIS (R2 NOT = 1) BY V 22 E 2 E. CONTINUE		
#	AFTER KEYING IN PROCEED.		
#			
#	4. BOTH VEHICLE STATE VECTORS UPDATED BY CONIC EQS.		
#			
#	5. HALF MAGNITUDE UNIT LOS VECTOR (IN STABLE MEMBER COORDINATES) AND		

EXTENDED_VERBS

```
1 # HALF MAGNITUDE UNIT SPACECRAFT AXIS VECTOR (IN BODY COORDINATES)
2 # PREPARED FOR VECPOINT.
3
4 #
5 # 6. GIMBAL ANGLES FROM VECPOINT TRANSFORMED INTO FDAI BALL ANGLES BY
6 # BALLANGS. FLASH DISPLAY V 06 N 18 AND AWAIT RESPONSE.
7
8 #
9 # 7. RECYCLE -- RETURN TO STEP 4.
10 # TERMINATE -- EXIT R63.
11 # PROCEED -- RESET 3AXISFLG AND CALL R60LEM FOR ATTITUDE MANEUVER.
12
13 V89PERF TC CHKPOOH
14 TC TESTXACT
15 CAF PRI010
16 TC FINDVAC
17 EBANK= RONE
18 2CADR V89CALL
19
20 TC ENDOFJOB
21
22 # V90PERF VERB 90 DESCRIPTION
23 # REQUEST RENDEZVOUS OUT-OF-PLANE DISPLAY (R36)
24 #
25 # 1. SET EXT VERB DISPLAY BUSY FLAG.
26 # 2. SCHEDULE R36 CALL WITH PRIORITY 10
27 # A. DISPLAY
28 # TIME OF EVENT -- HOURS, MINUTES, SECONDS
29 # Y OUT-OF-PLANE POSITION -- NAUTICAL MILES
30 # YDOT OUT-OF-PLANE VELOCITY -- FEET/SECOND
31 # PSI ANGLE BTW LINE OF SIGHT AND FORWARD
32 # DIRECTION VECTOR IN HORIZONTAL PLANE -- DEGREES
33
34 V90PERF TC TESTXACT
35 CAF PRI07 # R36,V90
36 TC FINDVAC
37 EBANK= RPASS36
38 2CADR R36
39
40 TCF ENDOFJOB
41
42 # MINIMP VERB 76 DESCRIPTION
43 # MINIMUM IMPULSE MODE
44 #
45 # 1. SET MINIMUM IMPULSE RHO MODE FLAG TO 1.
46
47 MINIMP INHINT
48 CS DAPBOOLS
49 MASK PULSES # PULSES = 1 INDICATES MIN IMP MODE
50 ADS DAPBOOLS
51 TCF GOPIN # RETURN VIA PINBRNCH
52
53 # NOMINIMP VERB 77 DESCRIPTION
54 # RATE COMMAND MODE
```



1					1
2					2
3					3
4	#	1.	SET MINIMUM IMPULSE RHO MODE FLAG TO 0. (ZERO INDICATES NOT MINIMUM IMPULSE MODE.).		5
5	#	2.	MOVE CDUX, CDUY, CDUZ INTO CDUXD, CDUYD, CDUZD.		6
6					7
7					8
8	NOMINIMP	INHINT			9
9		CS	PULSES		10
10		MASK	DAPBOOLS		11
11		TS	DAPBOOLS	# PULSES = NOT IN MINIMUM UMPULSE MODE	12
12		TC	IBNKCALL		13
13		CADR	ZATTEROR		14
14		TC	GOPIN		15
15					16
16					17
17					18
18					19
19					20
20					21
21					22
22					23
23					24
24					25
25					26
26					27
27					28
28					29
29					30
30					31
31					32
32					33
33					34
34					35
35					36
36					37
37					38
38					39
39					40
40					41
41					42
42					43
43					44
44					45
45					46
46					47
47					48
48					49
49					50
50					51
51					52
52					53
53					54
54					55
55					56
56					57
57					58
58					59
59					60
60					61

EXTENDED_VERBS

1						1
2	#	CREMANU	VERB 49	DESCRIPTION		2
3	#		START AUTOMATIC	ATTITUDE	MANEUVER	3
4	#					4
5	#	1.	REQUIRE PROGRAM 00 ACTIVE.			5
6	#	2.	SET EXT VERB DISPLAY BUSY FLAG.			6
7	#	3.	SCHEDULE R62DISP WITH PRIORITY 10.			7
8	#	4.	RELEASE EXT VERB DISPLAY.			8
9	#					9
10	#	R62DISP				10
11	#	1.	DISPLAY FLASHING V06,N22.			11
12	#		RESPONSES			12
13	#		A.	TERMINATE		13
14	#			1.	GOTOP00H	14
15	#		B.	PROCEED		15
16	#			1.	SET 3AXISFLG TO INDICATE MANEUVER IS SPECIFIED BY 3 AXIS.	16
17	#			2.	EXECUTE R60LEM (ATTITUDE MANEUVER).	17
18	#		C.	ENTER		18
19	#			1.	REPEAT FLASHING V06,N22.	19
20						20
21	CREWMANU	TC	CHKP00H	#	DEMAND P00	21
22						22
23			TC	TESTXACT		23
24						24
25			CAF	PRI010		25
26			TC	FINDVAC		26
27			EBANK=	BCDU		27
28			2CADR	R62DISP		28
29						29
30			TC	ENDOFJOB		30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60


```
# TRMTRACK      VERB 56      DESCRIPTION
#      TERMINATE TRACKING (P20 AND P25).

#
#      1.      KNOCK DOWN RENDEZVOUS, TRACK, AND UPDATE FLAGS.
#      2.      REQUIRE P20 OR P25 NOT RUNNING ALONE OR GO TO GOGOPOOH (REQUEST PROGRAM 00).
#      3.      SCHEDULE V56TOVAC WITH PRIORITY 30.

#      V56TOVAC
#      1.      EXECUTE INTSTALL (IF INTEGRATION IS RUNNING, STALL UNTIL IT IS FINISHED.).
#      2.      ZERO GROUP 2 TO HALT P20.
#      3.      TRANSFER CONTROL TO GOPROG2 (SOFTWARE RESTART).

TRMTRACK      CA      BITS9+7      # IS REND OR P25 FLAG ON
              MASK    FLAGWRD0

              EXTEND
              BZF      GOPIN      # NO

              TC      DOWNFLAG
              ADRES    RNDVZFLG

              TC      DOWNFLAG
              ADRES    P25FLAG

              TC      DOWNFLAG      # ENSURE SEARCH FLAG IS OFF
              ADRES    SRCHOPTN

              CA      TRACKBIT      # IS TRACK FLAG ON?
              MASK    FLAGWRD1
              EXTEND

              BZF      GOPIN

              TC      POSTJUMP
              CADR     TRMTRAK1

BITS9+7      OCT      500

              SETLOC  SBAND      # BANK 42
              BANK

              COUNT*  $$/EXTVB

TRMTRAK1     TC      DOWNFLAG
              ADRES    UPDATFLG      # UPDATE FLAG DOWN
              TC      DOWNFLAG

              ADRES    TRACKFLG      # TRACK FLAG DOWN
              TC      DOWNFLAG
              ADRES    IMUSE

              TC      INTPRET
              CALL

              INTSTALL      # DON'T INTERRUPT INTEGRATION
```

```
1
2      EXIT
3
4      TC      PHASCHNG
5      OCT      2          # KILL GROUP 2 TO HALT P20 ACTIVITY
6
7      INHINT
8      TC      IBNKCALL    # ZERO THE COMMANDED RATES TO STOP
9      CADR     STOPRATE   # MANEUVER
10
11     TC      IBNKCALL
12     CADR     RESTORDB
13
14     TC      CLRADMOD     # CLEAR BITS 10 + 15 OF RADMODES.
15
16     CS      BIT14        # DISABLE LOCKON
17     EXTEND
18     WAND     CHAN12
19     TC      POSTJUMP
20     CADR     GOPROG2     # CAUSE RESTART.
21
22     # DNEDUMP      VERB 74      DESCRIPTION
23     #      INITIALZE DOWN-TELEMETRY PROGRAM FOR ERASABLE MEMORY DUMP.
24     #
25     #      1.      SET EXT VERB DISPLAY BUSY FLAG.
26     #      2.      REPLACE CURRENT DOWNLIST WITH ERASABLE MEMORY.
27     #      3.      RELEASE EXT VERB DISPLAY.
28
29     SETLOC   EXTVERBS
30     BANK
31
32     COUNT*   $$/EXTVB
33
34     DNEDUMP  EBANK= 400
35             CAF     LDNDUMPI
36             TS      DNTMGOTO
37             TC      GOPIN
38
39     V74      EQUALS  DNEDUMP
40     LDNDUMPI REMADR  DNDUMPI
41
42     # LEMVEC   VERB 80      DESCRIPTION
43     #      UPDATE LEM STATE VECTOR
44     #      RESET VHUPFLG TC 0
45
46     LEMVEC   TC      DOWNFLAG
47             ADRES    VEHUPFLG    # VB 80 -- VEHUPFLG DOWN INDICATES LEM
48
49             TC      NOUPDOWN
50
51     # CSMVEC   VERB 81      DESCRIPTION
52     #      UPDATE CSM STATE VECTOR
```



1	#				SET VEHUPFLG TO 1	1
2						2
3						3
4	CSMVEC	TC	UPFLAG			4
5		ADRES	VEHUPFLG	# VB 81 -- VEHUPFLG UP INDICATES CSM		5
6						6
7	NOUPDOWN	TC	DOWNFLAG			7
8		ADRES	NOUPFLAG			8
9						9
10		TCF	GOPIN			10
11						11
12	# UPDATOFF	VERB 95	DESCRIPTION			12
13	# INHIBIT	STATE VECTOR UPDATES BY INCORP				13
14	#	SET NOUPFLAG TO 1				14
15						15
16	UPDATOFF	TC	UPFLAG	# VB 95 SET NOUPFLAG		16
17		ADRES	NOUPFLAG			17
18						18
19		TC	GOPIN			19
20						20
21						21
22						22
23						23
24						24
25						25
26						26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

#	SYSTEST	VERB 92	DESCRIPTION
#	OPERATE	IMU	PERFORMANCE TEST.
#			
#	1.	REQUIRE	PROGRAM 00 OR TURN ON OPERATOR ERROR.
#	2.	SET EXT	VERB BUSY FLAG.
		EBANK=	QPLACE
SYSTEST	TC	CHKPOOH	# DEMAND P00
	TC	TESTXACT	
	CAF	PRI022	
	TC	FINDVAC	
	EBANK=	QPLACE	
	SBANK=	IMUSUPER	
	2CADR	REDO	
	TC	ENDOFJOB	
# VERB 93			CLEAR RENDWFLG, CAUSES W-MATRIX TO BE RE-INITIALIZED.
WMATRXNG	INHINT		
	CS	RENDWBIT	
	MASK	FLAGWRD5	
	TS	FLAGWRD5	
	TC	GOPIN	
GOSHOSUM	EQUALS	SHOWSUM	
SHOWSUM	TC	CHKPOOH	# *
	TC	TESTXACT	# *
	CAF	PRI07	# * ALLOW OTHER CHARINS.
	TC	PRI0CHNG	# *
	CAF	S+1	# *
	TS	SKEEP6	# * SHOWSUM OPTION
	CAF	S+ZERO	# *
	TS	SMODE	# * TURN OFF SELF-CHECK
	CA	SELFADRS	# *
	TS	SELFRET	# *
	TC	STSHOSUM	# * ENTER ROPECHK
SDISPLAY	LXCH	SKEEP2	# * BANK # FOR DISPLAY
	LXCH	SKEEP3	# * BUGGER WORD FOR DISPLAY
NOKILL	CA	ADRS1	# *
	TS	MPAC +2	# *
	CA	VNCON	# * 0501
	TC	BANKCALL	# *
	CADR	GOXDSPF	# *
	TC	+3	# *
	TC	NXTBNK	# *

1412THE

#	DAPDISP	VERB 48	DESCRIPTION
#	LOAD	AUTO PILOT DATA	
#			
#	1.	REQUIRE	EXT VERB DISPLAY AVAILABLE AND SET BUSY FLAG.
#	2.	EXECUTE	DAPDATA1, DAPDATA2, AND DAPDATA3.
#	3.	RELEASE	EXT VERB DISPLAY SYSTEM.
DAPDISP	TC	TESTXACT	
	CAF	PRI07	# R03
	TC	PRI0CHNG	
	TC	POSTJUMP	
	CADR	DAPDATA1	
	BANK	34	
	SETLOC	LOADDAP	
	BANK		
	COUNT*	\$\$/R03	
	SBANK=	LOWSUPER	# FOR SUBSEQUENT LOW 2CADR'S
DAPDATA1	CAF	BOOLSMSK	# SET DISPLAY ACCORDING TO DAPBOOLS BITS.
	MASK	DAPBOOLS	# LM
	TS	DAPDATR1	# LM
	CS	FLGWRD10	# SET BIT 14 TO BE COMPLEMENT OF APSFLAG.
	MASK	APSFLBIT	
	CCS	A	
	CAF	BIT14	
	ADS	DAPDATR1	
CHKDATA1	CAE	DAPDATR1	# IF BITS 13 AND 14 ARE BOTH ZERO, FORCE
	MASK	BIT13-14	# A ONE INTO BIT 13.
	EXTEND		
	BZF	FORCEONE	
	CAE	DAPDATR1	# ENSURE THAT NO ILLEGAL BITS SET BY CREW.
MSKDATR1	MASK	DSPLYMSK	
	TS	DAPDATR1	
	CAF	VO1N46	# LM
	TC	BANKCALL	
	CADR	GOXDSPFR	
	TCF	ENDEXT	# V34E TERMINATE
	TCF	DPDAT1	# V33E PROCEED
	TCF	CHKDATA1	# E NEW DATA CHECK AND REDISPLAY
	CAF	REVCNT	# BITS 2 & 3: BLANKS R2 & R3.
	TC	BLANKET	
	TCF	ENDOFJOB	
FORCEONE	CAF	BIT13	
	ADS	DAPDATR1	
	TCF	MSKDATR1	
DPDAT1	INHINT		# INHINT FOR SETTING OF FLAG BITS AND MASS
	CS	APSFLBIT	# ON BASIS OF DISPLAYED DAPDATR1.
	MASK	FLGWRD10	
	TS	L	# SET APSFLAG TO BE COMPLEMENT OF BIT 14.

1					
2		CS	DAPDATR1		
3		MASK	BIT14		
4		CCS	A		
5		CAF	APSFLBIT		
6		AD	L		
7		TS	FLGWRD10		
8		CS	DAPDATR1	# SET BITS OF DAPBOOLS ON BASIS OF DISPLAY	
9		MASK	BIT13-14	# MASK OUT CSMDOCKD (BIT 13) UNLESS BOTH	
10		CCS	A	# 13 AND 14 ARE SET.	
11		CS	CSMDOCKD		
12		AD	BOOLSMSK		
13		MASK	DAPDATR1		
14		TS	L		
15		CS	BOOLSMSK		
16		MASK	DAPBOOLS		
17		AD	L		
18		TS	DAPBOOLS		
19		MASK	CSMDOCKD	# LOAD MASS IN ACCORDANCE WITH CSMDOCKD.	
20		CCS	A	# MASS IS USUALLY OKAY, SO DO	
21		CAE	CSMMASS	# NOT TOUCH ITS LOW-ORDER PART.	
22		AD	LEMMASS		
23		TS	MASS		
24		CAE	DAPBOOLS		
25		MASK	ACC4OR2X	# 2 OR 4 JET X-TRANSLATION	
26		EXTEND		# (BIT ACC4OR2X = 1 FOR 4 JETS)	
27		BZF	+5		
28		CS	BIT15		
29		MASK	FLAGWRD1	# CLEAR NJTSFLAG TO 0 FOR 4 JETS	
30		TS	FLAGWRD1		
31		TCF	+4		
32		CS	FLAGWRD1	# SET NJTSFLAG TO 1 FOR 2 JETS	
33		MASK	BIT15		
34		ADS	FLAGWRD1		
35		CA	DAPBOOLS	# SELECT DESIRED KALCMANU AUTOMATIC	
36		MASK	THREE	# MANEUVER RATE	
37		DOUBLE		# RATEINDX HAS TO BE 0,2,4,6 SINCE RATES	
38		TS	RATEINDX	# ARE DP	
39		TC	POSTJUMP		
40		CADR	STIKLOAD		
41					
42	VO1N46	VN	0146		
43	DSPLYMSK	OCT	33113		
44	BOOLSMSK	OCT	13113		
45					
46		BANK	01		
47		SETLOC	LOADDAP1		
48		BANK			
49					
50		COUNT*	\$\$/R03		
51					
52	STIKLOAD	CAF	EBANK6		
53					
54					
55					
56					
57					
58					
59					
60					

1				
2		TS	EBANK	
3		EBANK=	STIKSENS	
4		CA	RHCSALE	# SET STICK SENSITIVITY TO CORRESPOND TO A
5		MASK	DAPBOOLS	# MAXIMUM COMMANDED RATE (AT 42 COUNTS) OF
6		CCS	A	# 20 D/S (NORMAL) OR 4 D/S (FINE), SCALED
7		CA	NORMAL	# AT 45 D/S.
8		AD	FINE	
9		TS	STIKSENS	
10		CA	-0.6D/S	
11		TS	-RATEDB	# LM-ONLY BREAKOUT LEVEL IS .6 D/S.
12		CA	CSMDOCKD	# IF CSM-DOCKED, DIVIDE STICK SENSITIVITY
13		MASK	DAPBOOLS	# BY 10. NORMAL SCALING IS THEN 2 D/S AND
14		EXTEND		# FINE SCALING IS 0.4 D/S
15		BZF	+7	# BRANCH IF CSM IS NOT DOCKED.
16		CA	STIKSENS	
17		EXTEND		
18		MP	1/10	
19		TS	STIKSENS	
20		CA	-0.3D/S	# CSM-DOCKED BREAKOUT LEVEL IS .3 D/S.
21		TS	-RATEDB	
22		RELINT		# PROCEED TO NOUN 47, MASS LOAD.
23				
24	DAPDATA2	CAF	V0647	
25		TC	BANKCALL	
26		CADR	GOXDSPFR	
27		TCF	ENDR03	# V34E TERMINATE. FIRST SET DB. DO 1/ACCS
28		TCF	DAPDAT2	# V33E PROCEED
29		TCF	DAPDATA2	# LOAD NEW DATA AND RECYCLE
30		CAF	BIT3	# BLANKS R3
31		TC	BLANKET	# LM
32		TCF	ENDOFJOB	
33	ENDR03	INHINT		
34		TC	IBNKCALL	
35		CADR	RESTORDB	
36		TCF	ENDEXT	# DOES RELINT
37				
38	DAPDAT2	CS	FLGWRD10	# DETERMINE STAGE FROM APSFLAG
39		MASK	APSFLBIT	
40		CCS	A	
41		CA	MINLMD	
42		AD	MINMINLM	
43		AD	LEMMASS	# LEMMASS MUST BE GREATER THAN EMPTY LEM
44		EXTEND		
45		BZMF	DAPDATA2	# ASK FOR NEW MASSES
46		CAE	DAPBOOLS	
47		MASK	CSMDOCKD	
48		EXTEND		
49		BZF	LEMALONE	# SKIP TEST ON CSMMASS IF NOT DOCKED.
50		CS	MINCSM	# TEST CSM MASS
51		AD	CSMMASS	# CSMMASS MUST BE GREATER THAN EMPTY CSM
52				
53				
54				
55				
56				
57				
58				
59				
60				

	EXTEND		
	BZMF	DAPDATA2	# ASK FOR NEW MASSES
LEMALONE	CAE	CSMMASS	# DOCKED: MASS = CSMMASS + LEMMASS
	AD	LEMASS	# LEM ALONE: MASS = LEMMASS
	ZL		
	DXCH	MASS	
	INHINT		
	TC	IBNKCALL	# SET DEADBANK AND COMPUTE MOMENTS OF
	CADR	RESTORDB	# INERTIA.
	RELINT		# PROCEED TO NOUN 48 (OR END).
DAPDATA3	CS	FLGWRD10	
	MASK	APSFLBIT	
	EXTEND		# END ROUTINE IF LEM HAS STAGED.
	BZF	ENDEXT	
	CAF	V06N48	# DISPLAY TRIM ANGLES AND REQUEST RESPONSE
	TC	BANKCALL	
	CADR	GOXDSPFR	
	TC	ENDEXT	
	TCF	DPDAT3	# V33E GO DO TRIM (WAITLIST TO TRIMGIMB)
	TCF	-5	# LOAD NEW DATA AND RECYCLE
	CAF	BIT3	
	TC	BLANKET	# BLANK R3
DPDAT3	TCF	ENDOFJOB	
	CAF	BIT1	# GO TO TRIMGIMB VIA WAITLIST SO IT
	INHINT		# CAN USE FIXDELAY AND VARDELAY
	TC	WAITLIST	
	EBANK=	ROLLTIME	
	2CADR	TRIMGIMB	
TRIMDONE	TCF	ENDOFJOB	# DOES A RELINT
	CAF	V50N48	
	TC	BANKCALL	# TRIM IS FINISHED; PLEASE TERMINATE R03
	CADR	GOMARK3R	
	TC	ENDEXT	# V34E TERMINATE
	TC	ENDEXT	
	TC	ENDEXT	
	CAF	OCT24	# BIT5 TO CHANGE TO PERFORM, 3 TO BLANK 43
	TC	BLANKET	
	TCF	ENDOFJOB	
V0647	VN	0647	
V06N48	VN	0648	
V50N48	VN	5048	
NORMAL	DEC	.660214	
			# NORMAL SCALING IS 20 D/S
FINE	DEC	.165054	# FINE STICK SCALING (4 D/S).
1/10	DEC	.1	# FACTOR FOR CSM-DOCKED SCALING
-0.6D/S	DEC	-218	

-0.3D/S	DEC	-109
---------	-----	------

VERB 66 VEHICLES ARE ATTACHED. MOVE THIS VEHICLE STATE VECTOR TO
OTHER VEHICLE STATE VECTOR.

USE SUBROUTINE GENTRAN.

BANK 7
SETLOC EXTVERBS
BANK

COUNT* \$\$/EXTVB

EBANK= RRECTHIS

ATTACHED CAF PRI010
 TC FINDVAC
 EBANK= RRECTHIS

2CADR ATTACHIT

TC ENDOFJOB

ATTACHIT TC INTPRET
 CALL

SET INTSTALL
 BON
 MOONOTH
 MOONTHIS
 +3

CLEAR

 MOONOTH

EXIT
CAF OCT51
TC GENTRAN
ADRES RRECTHIS # OUR STATE VECTOR INTO OTHER VIA GENTRAN
ADRES RRECTOTH

RELINT
TC INTPRET

CALL # UPDATE R-OTHER, V-OTHER

LXA,2 PTOALEM
 CALL
 PBODY
 SVDWN1

EXIT

CAF TCPINAD
INDEX FIXLOC

TS QPRET
TC POSTJUMP
CADR INTWAKE # FREE INTEGRATION AND EXIT.

```
1
2
3 TCPIN          RTB
4                PINBRNCH
5
6 OCT51          OCT    51
7 TCPINAD        CADR   TCPIN
8
9 # VERB 96      SET QUITFLAT TO STOP INTEGRATION.
10 #
11 #            GO TO V37 WITH ZERO TO CAUSE P00.
12 #            STATEINT WILL CHECK QUITFLAG AND SKIP 1ST PASS,
13 #            THUS ALLOWING A 10 MINUT PERIOD WITHOUT INTEGRATION.
14
15 VERB96         TC      UPFLAG      # QUITFLAG WILL CAUSE INTEGRATION TO EXIT
16                ADRES   QUITFLAG    #            AT NEXT TIMESTEP
17
18                CAF     ZERO
19                TC      POSTJUMP
20                CADR    V37          # GO TO P00
21
22 # VERB 67:     DISPLAY OF W MATRIX
23
24 V67            TC      TESTXACT
25                CAF     PRI05
26                TC      FINDVAC
27                EBANK=   WWPOS
28                2CADR    V67CALL
29
30                TC      ENDOFJOB
31
32 # VERB 65      DISABLE U,V JETS DURING DPS BURNS
33
34 SNUFFOUT       TC      UPFLAG
35                ADRES   SNUFFER
36                TC      GOPIN
37
38 # VERB 75      ENABLE U,V JETS DURING DPS BURNS
39
40 OUTSNUFF       TC      DOWNFLAG
41                ADRES   SNUFFER
42                TC      GOPIN
43
44 # VERB 85      DISPLAY RR LOS AZIMUTH AND ELEVATION.
45 #
46 # AZIMUTH IS THE ANGLE BETWEEN THE LOS AND THE X-Z NB PLANE, 0-90 DEG IN THE +Y HEMISPHERE,
47 # 360-270 DEG IN THE -Y HEMISPHERE.
48 #
49 # ELEVATION IS THE ANGLE BETWEEN +ZNB AND THE PROJECTION OF THE LOS INTO THE X-Z PLANE, 0-360 ABOUT +Y.
50
51                EBANK=   RR-AZ
52 VERB85         TC      TESTXACT
```

1				
2				
3		TC	POSTJUMP	
4		CADR	DSPRRLOS	
5				
6		SETLOC	PINBALL1	
7		BANK		
8				
9		COUNT*	\$\$/EXTVB	
10				
11	DSPRRLOS	CAF	PRI05	
12		TC	FINDVAC	
13		EBANK=	RR-AZ	
14		2CADR	RRLOSDSP	
15				
16		CAF	PRI04	
17		TC	PRI0CHNG	
18		CAF	V16N56	
19		TC	BANKCALL	
20		CADR	GOMARKFR	
21		TC	B5OFF	
22		TC	B5OFF	
23		TC	B5OFF	
24				
25		CAF	BIT3	
26		TC	BLANKET	
27		TC	ENDOFJOB	
28				
29	RRLOSDSP	EXTEND		
30		DCA	CDUT	
31		DXCH	MPAC	
32		TC	INTPRET	
33		CALL		
34			RRNBMPAC	# GET RR LOS IN BODY AXIS.
35		STORE	0D	# UNIT LOS
36		STODL	6D	
37			HI6ZEROS	
38		STOVL	8D	
39			6D	
40		UNIT		
41		STORE	6D	# UNIT OF LOS PROJ IN X-Z PLANE
42		DOT		
43			UNITZ	
44		STOVL	COSTH	# 16D
45			UNITX	
46		DOT		
47			6D	
48		STCALL	SINTH	# 18D
49			ARCTRIG	
50		BPL	DAD	# INSURE DISPLAY OF 0-360 DEG.
51			+2	
52			DPPOS MAX	# INTRODUCES AND ERROR OF B-28 REVS.
53				
54				
55				
56				
57				
58				
59				
60				

1	# EXTENDED_VENDS			1
2				2
3		STOVL	RR-ELEV	3
4			OD	4
5		DOT		5
6			UNITY	6
7		STOVL	SINTH	7
8			OD	8
9		DOT		9
10			6D	10
11		STCALL	COSTH	11
12			ARCTRIG	12
13		BPL	DAD	13
14			+2	14
15			DPPOS MAX	15
16			# INSURE DISPLAY OF 0-360 DEG.	16
17				17
18			# INTRODUCES AN ERROR OF B-28 REVS.	18
19				19
20		STORE	RR-AZ	20
21		EXIT		21
22		CA	1SEC	22
23		TC	BANKCALL	23
24		CADR	DELAYJOB	24
25				25
26		CA	BIT5	26
27		MASK	EXTVBACT	27
28		CCS	A	28
29		TC	RRLOSDSP	29
30		TC	ENDEXT	30
31				31
32	V16N56	VN	1656	32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

THE FOLLOWING REFERS TO THE NOUN TABLES

#

COMPONENT CODE NUMBER INTERPRETATION

#

00000 1 COMPONENT

00001 2 COMPONENT

00010 3 COMPONENT

X1XXX BIT 4 = 1. DECIMAL ONLY

1XXXX BIT 5 = 1. NO LOAD

END OF COMPONENT CODE NUMBER

#

SF ROUTINE CODE NUMBER INTERPRETATION

#

00000 OCTAL ONLY

00001 STRAIGHT FRACTIONAL

00010 CDU DEGREES (XXX.XX)

00011 ARITHMETIC SF

00100 ARITH DP1 OUT (MULT BY 2EXP14 AT END) IN (STRAIGHT)

00101 ARITH DP2 OUT (STRAIGHT) IN (SL 7 AT END)

00110 LANDING RADAR POSITION (+0000X)

00111 ARITH DP3 OUT (SL 7 AT END) IN (STRAIGHT)

01000 WHOLE HOURS IN R1, WHOLE MINUES (MOD 60) IN R2,
SECONDS (MOD 60) OXX.XX IN R3. *** ALARMS IF USED WITH OCTAL

01001 MINUTES (MOD 60) IN D1D2, D3 BLANK, SECONDS (MOD 60) IN D4D5

LIMITS TO 59B59 IF MAG EXCEEDS THIS VALUE.

ALARMS IF USED WITH OCTAL ***** IN (ALARM)

01010 ARITH DP4 OUT (STRAIGHT) IN (SL 3 AT END)

01011 ARITH1 SF OUT (MULT BY 2EXP14 AT END) IN (STRAIGHT)

01100 2 INTEGERS IN D1D2, D4D5, D3 BLANK.

ALARMS IF USED WITH OCTAL ***** IN (ALARM)

01101 360-CDU DEGREES (XXX.XX)

#

END OF SF ROUTINE CODE NUMBERS

#

SF CONSTANT CODE NUMBER INTERPRETATION

#

00000 WHOLE USE ARITH

00000 DP TIME SEC (XXX.XX SEC) USE ARITHDP1

00000 LR POSITION (+0000X) USE LR POSITION

00001 SPARE

00010 CDU DEGREES USE CDU DEGREES

00010 360-CDU DEGREES USE 360-CDU DEGREES

00011 DP DEGREES (90 XX.XXX DEG USE ARITHDP3

00100 DP DEGREES (360) XXX.XX DEG USE ARITHDP4

00101 DEGREES (180) XXX.XX DEG USE ARITH

00101 OPTICAL TRACKER AZIMUTH ANGLE (XXX.XXDEG)

USE ARITHDP1

00110 WEIGHT2 (XXXXX. LBS) USE ARITH1

#	00111	POSITION5 (XXX.XX NAUTICAL MILES)	
#			USE ARITHDP3
#	01000	POSITION4 (XXXX.X NAUTICAL MILES)	
#			USE ARITHDP3
#	01001	VELOCITY2 (XXXXX. FT/SEC)	USE ARITHDP4
#	01010	VELOCITY3 (XXXX.X FT/SEC)	USE ARITHDP3
#	01011	ELEVATION DEGREES (89.999 MAX)	USE ARITH
#	01100	RENDEZVOUS RADAR RANGE (XXX.XX NAUT MI)	
#			USE ARITHDP1
#	01101	RENDEZVOUS RADAR RANGE RATE (XXXXX.FT/SEC)	
#			USE ARITHDP1
#	01110	LANDING RADAR ALTITUDE (XXXXX.FEET)	
#			USE ARITHDP1
#	01111	INITIAL/FINAL ALTITUDE (XXXXX. FEET)	
#			USE ARITHDP1
#	10000	ALTITUDE RATE (XXXXX.FT/SEC)	USE ARITH
#	10001	FORWARD/LATERAL VELOCITY (XXXXX.FEET/SEC)	
#			USE ARITH
#	10010	ROTATIONAL HAND CONTROLLER ANGLE RATES	
#		XXXXX.DEG/SEC	USE ARITH
#	10011	LANDING RADAR VELX (XXXXX.FEET/SEC)	
#			USE ARITHDP1
#	10100	LANDING RADAR VELY (XXXXX.FEET/SEC)	
#			USE ARITHDP1
#	10101	LANDING RADAR VELZ (XXXXX.FEET/SEC)	
#			USE ARITHDP1
#	10110	POSITION7 (XXXX.X NAUT MI)	USE ARITHDP4
#	10111	TRIM DEGREES2 (XXX.XX DEG)	USE ARITH
#	11000	COMPUTED ALTITUDE (XXXXX. FEET)	
#			USE ARITHDP1
#	11001	DP DEGREES (XXXX.X DEG)	USE ARITHDP3
#	11010	POSITION9 (XXXX.X FT)	USE ARITHDP3
#	11011	VELOCITY4 (XXXX.X FT/SEC)	USE ARITHDP2
#	11100	RADIANS (XXX.XXX RADIANS)	USE ARITHDP4
#			

END OF SF CONSTANT CODE NUMBERS

FOR GREATER THAN SINGLE PRECISION SCALES, PUT ADDRESS OF MAJOR PART INTO
NOUN TABLES.

OCTAL LOADS PLACE +0 INTO MAJOR PART, DATA INTO MINOR PART.

OCTAL DISPLAYS SHOW MINOR PART ONLY.

TO GET AT BOTH MAJOR AND MINOR PARTS (IN OCTAL), USE NOUN 01.

A NOUN MAY BE DECLARED :DECIMAL ONLY: BY MAKING BIT4=1 OF ITS COMPONENT
CODE NUMBER. IF THIS NOUN IS USED WITH ANY OCTAL DISPLAY VERB, OR IF
DATA IS LOADED IN OCTAL, IT ALARMS.# IN LOADING AN :HOURS, MINUTES, SECONDS: NOUN, ALL 3 WORDS MUST BE
LOADED, OR ALARM.



ALARM IF AN ATTEMPT IS MADE TO LOAD :SPLIT MINUTES/SECONDS: (MMBSS).
THIS IS USED FOR DISPLAY ONLY.

THE FOLLOWING ROUTINES ARE FOR READING THE NOUN TABLES AND THE SF TABLES
(WHICH ARE IN A SEPARATE BANK FROM THE REST OF PINBALL). THESE READING
ROUTINES ARE IN THE SAME BANK AS THE TABLES. THEY ARE CALLED BY DXCH Z.

LODNNTAB LOADS NNADTEM WITH THE NNADTAB ENTRY, NNTYPTTEM WITH THE
NNTYPTAB ENTRY. IF THE NOUN IS MIXED, IDADITEM IS LOADED WITH THE FIRST
IDADDTAB ENTRY, IDAD2TEM THE SECOND IDADDTAB ENTRY, IDAD3TEM THE THIRD
IDADDTAB ENTRY, RUTMXTEM WITH THE RUTMXTAB ENTRY. MIXBR IS SET FOR
MIXED OR NORMAL NOUN.

	BANK	6	
	SETLOC	PINBALL3	
	BANK		
	COUNT*	\$\$/NOUNS	
LODNNTAB	DXCH	IDAD2TEM	# SAVE RETURN INFO IN IDAD2TEM, IDAD3TEM.
	INDEX	NOUNREG	
	CAF	NNADTAB	
	TS	NNADTEM	
	INDEX	NOUNREG	
	CAF	NNTYPTAB	
	TS	NNTYPTTEM	
	CS	NOUNREG	
	AD	MIXCON	
	EXTEND		
	BZMF	LODMIXNN	# NOUN NUMBER G/E FIRST MIXED NOUN
	CAF	ONE	# NOUN NUMBER L/ FIRST MIXED NOUN
	TS	MIXBR	# NORMAL. +1 INTO MIXBR
	TC	LODNLV	
LODMIXNN	CAF	TWO	# MIXED. +2 INTO MIXBR.
	TS	MIXBR	
	INDEX	NOUNREG	
	CAF	RUTMXTAB -40D	# FIRST MIXED NOUN = 40.
	TS	RUTMXTEM	
	CAF	LOW10	
	MASK	NNADTEM	
	TS	Q	# TEMP
	INDEX	A	
	CAF	IDADDTAB	
	TS	IDAD1TEM	# LOAD IDAD1TEM WITH FIRST IDADDTAB ENTRY
	EXTEND		
	INDEX	Q	# LOAD IDAD2TEM WITH 2ND IDADDTAB ENTRY
	DCA	IDADDTAB +1	# LOAD IDAD3TEM WITH 3RD IDADDTAB ENTRY.
LODNLV	DXCH	IDAD2TEM	# PUT RETURN INFO INTO A, L.
	DXCH	Z	
MIXCON	=	OCT50	# (DEC 40)
# GTSFOUT LOADS SFTEMP1, SFTEMP2 WITH THE DP SFOUTAB ENTRIES.			
GTSFOUT	DXCH	SFTEMP1	# 2X(SFCONUM) ARRIVES IN SFTEMP1.

	EXTEND			
	INDEX	A		
SFCOM	DCA	SFOUTAB		
	DXCH	SFTEMP1		
	DXCH	Z		
# GTSFIN LOADS SFTEMP1, SFTEMP2 WITH THE DP SFINTAB INTRIES.				
GTSFIN	DXCH	SFTEMP1	# 2X(SFCONUM)	ARIVES IN SFTEMP1.
	EXTEND			
	INDEX	A		
	DCA	SFINTAB		
	TCF	SFCOM		
NNADTAB	OCT	00000	# NN	NORMAL NOUNS
	OCT	40000	# 00	NOT IN USE
	OCT	40000	# 01	SPECIFY MACHINE ADDRESS (FRACTIONAL)
	OCT	40000	# 02	SPECIFY MACHINE ADDRESS (WHOLE)
	OCT	40000	# 03	SPECIFY MACHINE ADDRESS (DEGREES)
	ECADR	DSPTM1	# 04	ANGULAR ERROR/DIFFERENCE
	ECADR	DSPTM1	# 05	ANGULAR ERROR/DIFFERENCE
	ECADR	OPTION1	# 06	OPTION CODE
	ECADR	XREG	# 07	ECADR OF WORD TO BE MODIFIED
			#	ONES FOR BITS TO BE MODIFIED
			#	1 TO SET OR 0 TO RESET SELECTED BITS
	ECADR	ALMCADR	# 08	ALARM DATA
	ECADR	FAILREG	# 09	ALARM CODES
	OCT	77776	# 10	CHANNEL TO BE SPECIFIED
	ECADR	TCSI	# 11	TIG OF CSI (HRS,MIN,SEC)
	ECADR	OPTIONX	# 12	OPTION CODE
			#	(USED BY EXTENDED VERBS ONLY)
	ECADR	TCDH	# 13	TIG OF CDH (HRS,MIN,SEC)
	ECADR	DSPTMX	# 14	CHECKLIST
			#	(USED BY EXTENDED VERBS ONLY)
	OCT	77777	# 15	INCREMENT MACHINE ADDRESS
	ECADR	DSPTMX	# 16	TIME OF EVENT (HRS,MIN,SEC)
	OCT	00000	# 17	SPARE
	ECADR	FDAIX	# 18	AUTO MANEUVER BALL ANGLES
	OCT	00000	# 19	SPARE
	ECADR	CDUX	# 20	ICDU ANGLES
	ECADR	PIPAX	# 21	PIPAS
	ECADR	THETAD	# 22	NEW ICDU ANGLES
	OCT	00000	# 23	SPARE
	ECADR	DSPTM2 +1	# 24	DELTA TIME FOR AGC CLOCK (HRS,MIN,SEC)
	ECADR	DSPTM1	# 25	CHECKLIST
			#	(USED WITH PLEASE PERFORM ONLY)
	ECADR	DSPTM1	# 26	PRIO/DELAY, ADRES, BBCON
	ECADR	SMODE	# 27	SELF TEST ON/OFF SWITCH

OCT	00000	# 28	SPARE
OCT	00000	# 29	SPARE
OCT	0	# 30	SPARE
OCT	0	# 31	SPARE
ECADR	-TPER	# 32	TIME TO PERIGEE (HRS,MIN,SEC)
ECADR	TIG	# 33	TIME OF IGNITION (HRS,MIN,SEC)
ECADR	DSPTM1	# 34	TIME OF EVENT (HRS,MIN,SEC)
ECADR	TTOGO	# 35	TIME TO GO TO EVENT (HRS,MIN,SEC)
ECADR	TIME2	# 36	TIME OF AGC CLOCK (HRS,MIN,SEC)
ECADR	TTPI	# 37	TIG OF TPI (HRS,MIN,SEC)
ECADR	TET	# 38	TIME OF STATE BEING INTEGRATED
OCT	00000	# 39	SPARE
# END OF NNADTAB FOR NORMAL NOUNS			
		# NN	MIXED NOUNS
OCT	64000	# 40	TIME TO IGNITION/CUTOFF
		#	VG
		#	DELTA V (ACCUMULATED)
OCT	02003	# 41	TARGET AZIMUTH
		#	ELEVATION
OCT	24006	# 42	APOGEE
		#	PERIGEE
		#	DELTA V (REQUIRED)
OCT	24011	# 43	LATITUDE
		#	LONGITUDE
		#	ALTITUDE
OCT	64014	# 44	APOGEE
		#	PERIGEE
		#	TFF
OCT	64017	# 45	MARKS
		#	TTI OF NEXT BURN
		#	MGA
OCT	00022	# 46	AUTOPILOT CONFIGURATION
OCT	22025	# 47	LEM WEIGHT
		#	CSM WEIGHT
OCT	22030	# 48	GIMBAL PITCH TRIM
		#	GIMBAL ROLL TRIM
OCT	24033	# 49	DELTA R
		#	DELTA V
		#	RADAR DATA SOURCE CODE
OCT	0	# 50	SPARE
OCT	22041	# 51	S-BAND ANTENNA PITCH
		#	YAW
OCT	00044	# 52	CENTRAL ANGLE OF ACTIVE VEHICLE
OCT	00000	# 53	SPARE
OCT	24052	# 54	RANGE
		#	RANGE RATE
		#	THETA
OCT	24055	# 55	NO. OF APSIDAL CROSSINGS

			#	ELEVATION ANGLE
			#	CENTRAL ANGLE
	OCT	02060	# 56	RR LOS AZIMUTH
			#	ELEVATION
	OCT	20063	# 57	DELTA R
	OCT	24066	# 58	PERIGEE ALT
			#	DELTA V TPI
			#	DELTA V TPF
	OCT	24071	# 59	DELTA VELOCITY LOS
	OCT	24074	# 60	HORIZONTAL VELOCITY
			#	ALTITUDE RATE
			#	COMPUTED ALTITUDE
	OCT	64077	# 61	TIME TO GO IN BRAKING PHASE
			#	TIME TO IGNITION
			#	CROSS RANGE DISTANCE
	OCT	64102	# 62	ABSOLUTE VALUE OF VELOCITY
			#	TIME TO IGNITION
			#	DELTA V (ACCUMULATED)
	OCT	24105	# 63	ABSOLUTE VALUE OF VELOCITY
			#	ALTITUDE RATE
			#	COMPUTED ALTITUDE
	OCT	64110	# 64	TIME LEFT FOR REDESIGNATION -- LPD ANGLE
			#	ALTITUDE RATE
			#	COMPUTED ALTITUDE
	OCT	24113	# 65	SAMPLED AGC TIME (HRS,MIN,SEC)
			#	(FETCHED IN INTERRUPT)
	OCT	62116	# 66	LR RANGE
			#	POSITION
	OCT	04121	# 67	LRVX
			#	LRVY
			#	LRVZ
	OCT	64124	# 68	SLANT RANGE TO LANDING SIGHT
			#	TIME TO GO IN BRAKING PHASE
			#	LR ALTITUDE -- COMPUTED ALTITUDE
	OCT	00000	# 69	SPARE
	OCT	04132	# 70	AOT DETENT CODE/STAR CODE
	OCT	04135	# 71	AOT DETENT CODE/STAR CODE
	OCT	02140	# 72	RR 360 -- TRUNNION ANGLE
			#	SHAFT ANGLE
	OCT	02143	# 73	NEW RR 360 -- TRUNNION ANGLE
			#	SHAFT ANGLE
	OCT	64146	# 74	TIME TO IGNITION
			#	YAWAFTER VEHICLE RISE
			#	PITCH AFTER VEHICLE RISE
	OCT	64151	# 75	DELTA ALTITUDE CDH
			#	DELTA TIME (CDH-CSI OR TPI-CDH)
			#	DELTA TIME (TPI-CDH OR TPI-NOMTPI)
	OCT	24154	# 76	DESIRED HORIZONTAL VELOCITY
			#	DESIRED RADIAL VELOCITY
			#	CROSS-RANGE DISTANCE

	OCT	62157	# 77	TIME TO ENGINE CUTOFF
			#	VELOCITY NORMAL TO CSM PLANE
	OCT	02162	# 78	RR RANGE
			#	RANGE RATE
	OCT	24165	# 79	CURSOR ANGLE
			#	SPIRAL ANGLE
			#	POSITION CODE
	OCT	02170	# 80	DATA INDICATOR
			#	OMEGA
	OCT	24173	# 81	DELTA V (LV)
	OCT	24176	# 82	DELTA V (LV)
	OCT	24201	# 83	DELTA V (BODY)
	OCT	24204	# 84	DELTA V (OTHER VEHICLE)
	OCT	24207	# 85	VG (BODY)
	OCT	24212	# 86	VG (LV)
	OCT	02215	# 87	BACKUP OPTICS LOS AZIMUTH
			#	ELEVATION
	OCT	24220	# 88	HALF UNIT SUN OR PLANET VECTOR
	OCT	24223	# 89	LANDMARK LATITUDE
			#	LONGITUDE/2
			#	ALTITUDE
	OCT	24226	# 90	Y
			#	Y DOT
			#	PSI
	OCT	04231	# 91	ALTITUDE
			#	VELOCITY
			#	FLIGHT PATH ANGLE
	OCT	00000	# 92	SPARE
	OCT	04237	# 93	DELTA GYRO ANGLES
	OCT	00000	# 94	SPARE
	OCT	0	# 95	SPARE
	OCT	0	# 96	SPARE
	OCT	04253	# 97	SYSTEM TEST INPUTS
	OCT	04256	# 98	SYSTEM TEST RESULTS
	OCT	24261	# 99	RMS IN POSITION
			#	RMS IN VELOCITY
			#	RMS IN BIAS
# END OF NNADTAB FOR MIXED NOUNS				
			# NN	NORMAL NOUNS
NNTYPTAB	OCT	00000	# 00	NOT IN USE
	OCT	04040	# 01	3COMP FRACTIONAL
	OCT	04140	# 02	3COMP WHOLE
	OCT	04102	# 03	3COMP CDU DEGREES
	OCT	00504	# 04	1COMP DPDEG(360)
	OCT	00504	# 05	1COMP DPDEG(360)
	OCT	04000	# 06	3COMP OCTAL ONLY
	OCT	04000	# 07	3COMP OCTAL ONLY
	OCT	04000	# 08	3COMP OCTAL ONLY

OCT	04000	# 09	3COMP	OCTAL ONLY
OCT	00000	# 10	1COMP	OCTAL ONLY
OCT	24400	# 11	3COMP	HMS (DEC ONLY)
OCT	02000	# 12	2COMP	OCTAL ONLY
OCT	24400	# 13	3COMP	HMS (DEC ONLY)
OCT	04140	# 14	3COMP	WHOLE
OCT	00000	# 15	1COMP	OCTAL ONLY
OCT	24400	# 16	3COMP	HMS (DEC ONLY)
OCT	0	# 17	SPARE	
OCT	04102	# 18	3COMP	CDU DEG
OCT	00000	# 19	SPARE	
OCT	04102	# 20	3COMP	CDU DEGREES
OCT	04140	# 21	3COMP	WHOLE
OCT	04102	# 22	3COMP	CDU DEGREES
OCT	00000	# 23	SPARE	
OCT	24400	# 24	3COMP	HMS (DEC ONLY)
OCT	04140	# 25	3COMP	WHOLE
OCT	04000	# 26	3COMP	OCTAL ONLY
OCT	00140	# 27	1COMP	WHILE
OCT	00000	# 28	SPARE	
OCT	00000	# 29	SPARE	
OCT	0	# 30	SPARE	
OCT	0	# 31	SPARE	
OCT	24400	# 32	3COMP	HMS (DEC ONLY)
OCT	24400	# 33	3COMP	HMS (DEC ONLY)
OCT	24400	# 34	3COMP	HMS (DEC ONLY)
OCT	24400	# 35	3COMP	HMS (DEC ONLY)
OCT	24400	# 36	3COMP	HMS (DEC ONLY)
OCT	24400	# 37	3COMP	HMS (DEC ONLY)
OCT	24400	# 38	3COMP	HMS (DEC ONLY)
OCT	00000	# 39	SPARE	

END OF NNTYPTAB FOR NORMAL NOUNS

		# NN	MIXED NOUNS
OCT	24500	# 40	3COMP MIN/SEC, VEL3, VEL3 (NO LOAD, DEC ONLY)
OCT	00542	# 41	2COMP CDU DEG, ELEV DEG
OCT	24410	# 42	3COMP POS4, POS4, VEL3 (DEC ONLY)
OCT	20204	# 43	3COMP DPDEG(360), DPDEG(360) POS4 (DEC ONLY)
OCT	00410	# 44	3COMP POS4, POS4, MIN/SEC (NO LOAD, DEC ONLY)
OCT	10000	# 45	3COMP WHOLE, MIN/SEC, DPDEG(360) (NO LOAD, DEC ONLY)
OCT	00000	# 46	1COMP OCTAL ONLY
OCT	00306	# 47	2COMP WEIGHT2 FOR EACH (DEC ONLY)
OCT	01367	# 48	2COMP TRIM DEG2 FOR EACH

			#		(DEC ONLY)
	OCT	00510	# 49	3COMP	POS4, VEL3, WHOLE
			#		(DEC ONLY)
	OCT	0	# 50	SPARE	
	OCT	00204	# 51	2COMP	DPDEG(360), DPDEG(360)
			#		(DEC ONLY)
	OCT	00004	# 52	1COMP	DPDEG(360)
	OCT	00000	# 53	SPARE	
	OCT	10507	# 54	3COMP	POS5, VEL3, DPDEG(360)
			#		(DEC ONLY)
	OCT	10200	# 55	3COMP	WHOLE, DPDEG(360), DPDEG(360)
			#		(DEC ONLY)
	OCT	00204	# 56	2COMP	DPDEG(360), DPDEG(360)
	OCT	00010	# 57	1COMP	POS4
			#		(DEC ONLY)
	OCT	24510	# 58	3COMP	POS4, VEL3, VEL3
			#		(DEC ONLY)
	OCT	24512	# 59	3COMP	VEL3 FOR EACH
			#		(DEC ONLY)
	OCT	60512	# 60	3COMP	VEL3, VEL3, COMP ALT
			#		(DEC ONLY)
	OCT	54000	# 61	3COMP	MIN/SEC, MIN/SEC, POS7
			#		(NO LOAD, DEC ONLY)
	OCT	24012	# 62	3COMP	VEL3, MIN/SEC, VEL3
			#		(NO LOAD, DEC ONLY)
	OCT	60512	# 63	3COMP	VEL3, VEL3, COMP ALT
			#		(DEC ONLY)
	OCT	60500	# 64	3COMP	2INT, VEL3, COMP ALT
			#		(NO LOAD, DEC ONLY)
	OCT	00000	# 65	3COMP	HMS (DEC ONLY)
	OCT	00016	# 66	2COMP	LANDING RADAR ALT, POSITION
			#		(NO LOAD, DEC ONLY)
	OCT	53223	# 67	3COMP	LANDING RADAR VELX, Y, Z
	OCT	60026	# 68	3COMP	POS7, MIN/SEC, COMP ALT
			#		(NO LOAD, DEC ONLY)
	OCT	00000	# 69	SPARE	
	OCT	0	# 70	3COMP	OCTAL ONLY FOR EACH
	OCT	0	# 71	3COMP	OCTAL ONLY FOR EACH
	OCT	00102	# 72	2COMP	360-CDU DEG, CDU DEG
	OCT	00102	# 73	2COMP	360-CDU DEG, CDU DEG
	OCT	10200	# 74	3COMP	MIN/SEC, DPDEG(360), DPDEG(360)
			#		(NO LOAD, DEC ONLY)
	OCT	00010	# 75	3COMP	POS4, MIN/SEC, MIN/SEC
			#		(NO LOAD, DEC ONLY)
	OCT	20512	# 76	3COMP	VEL3, VEL3, POS4
			#		(DEC ONLY)
	OCT	00500	# 77	2COMP	MIN/SEC, VEL3
			#		(NO LOAD, DEC ONLY)
	OCT	00654	# 78	2 COMP	RR RANGE, RR RANGE RATE
	OCT	00102	# 79	3COMP	CDU DEG, CDU DEG, WHOLE

1	# FINDALL_NOUN_TABLES					1
2			#		(DEC ONLY)	2
3		OCT 00200	# 80	2COMP	WHOLE, DPDEG(360)	3
4		OCT 24512	# 81	3COMP	VEL3 FOR EACH	4
5			#		(DEC ONLY)	5
6		OCT 24512	# 82	3COMP	VEL3 FOR EACH	6
7			#		(DEC ONLY)	7
8		OCT 24512	# 83	3COMP	VEL3 FOR EACH	8
9			#		(DEC ONLY)	9
10		OCT 24512	# 84	3COMP	VEL3 FOR EACH	10
11			#		(DEC ONLY)	11
12		OCT 24512	# 85	3COMP	VEL3 FOR EACH	12
13			#		(DEC ONLY)	13
14		OCT 24512	# 86	3COMP	VEL3 FOR EACH	14
15			#		(DEC ONLY)	15
16		OCT 00102	# 87	2COMP	CDU DEG FOR EACH	16
17		OCT 0	# 88	3COMP	FRAC FOR EACH	17
18			#		(DEC ONLY)	18
19		OCT 16143	# 89	3COMP	DPDEG(90), DPDEG(90), POS5	19
20			#		(DEC ONLY)	20
21		OCT 10507	# 90	3COMP	POS5, VEL3, DEPDEG(360)	21
22			#		(DEC ONLY)	22
23		OCT 10450	# 91	3COMP	POS4, VEL2, DPDEG(360)	23
24		OCT 00000	# 92	SPARE		24
25		OCT 06143	# 93	3COMP	DPDEG(90) FOR EACH	25
26		OCT 00000	# 94	SPARE		26
27		OCT 0	# 95	SPARE		27
28		OCT 0	# 96	SPARE		28
29		OCT 00000	# 97	3COMP	WHOLE FOR EACH	29
30		OCT 00000	# 98	3COMP	WHOLE, FRAC, WHOLE	30
31		OCT 71572	# 99	3COMP	POS9, VEL4, RADIANS	31
32			#		(DEC ONLY)	32
33						33
34	# END OF NNTYPTAB FOR MIXED NOUNS					34
35						35
36	SFINTAB	OCT 00006	# WHOLE, DP TIME (SEC)			36
37		OCT 03240				37
38		OCT 00000	# SPARE			38
39		OCT 00000				39
40		OCT 00000	# CDU DEGREES, 360-CDU DEGREES			40
41		OCT 00000	# (SFCONS IN DEGINSF)			41
42		OCT 10707	# DP DEGREES (90)			42
43		OCT 03435	# UPped BY 1			43
44		OCT 13070	# DP DEGREES (360) (POINT BETWN BITS 11-12)			44
45		OCT 34345	# UPped BY 1			45
46		OCT 00005	# DEGREES (180)			46
47		OCT 21616				47
48		OCT 26113	# WEIGHT2			48
49		OCT 31713				49
50		OCT 00070	# POSITION5			50
51		OCT 20460				51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

OCT	01065	# POSITION4
OCT	05740	
OCT	11414	# VELOCITY2 (POINT BETWN BITS 11-12)
OCT	31463	
OCT	07475	# VELOCITY3
OCT	16051	
OCT	00001	# ELEVATION DEGREES
OCT	03434	
OCT	00047	# RENDEZVOUS RADAR RANGE
OCT	21135	
OCT	77766	# RENDESVOUS RADAR RANGE RATE
OCT	50711	
2DEC*	.9267840599 E5 B-28*	# LANDING RADAR ALTITUDE
OCT	00002	# INITIAL/FINAL ALTITUDE
OCT	23224	
OCT	00014	# ALTITUDE RATE
OCT	06500	
OCT	00012	# FORWARD/LATERAL VELOCITY
OCT	36455	
OCT	04256	# ROT HAND CONT ANGLE RATE
OCT	07071	
2DEC*	-1.552795030 E5 B-28*	# LANDING RADAR VELX
2DEC*	.8250825087 E5 B-28*	# LANDING RADAR VELY
2DEC*	1.153668673 E5 B-28*	# LANDING RADAR VELZ
OCT	04324	# POSITION7
OCT	27600	
OCT	00036	# TRIM DEGREES2
OCT	20440	
OCT	00035	# COMPUTED ALTITUDE
OCT	30400	
OCT	23420	# DP DEGREES
OCT	00000	
2DEC	30480 B-19	# POSITION 9
2DEC	30.48 B-7	# VELOCITY4
2DEC	100 B-8	# RADIANS
		# END OF SFINTAB
SFOUTAB	OCT 05174	# WHOLE, DP TIME (SEC)
	OCT 13261	
	OCT 00000	# SPARE
	OCT 00000	
	OCT 00000	# CDU DEGREES, 360-CDU DEGREES

1				
2	OCT	00000	#	(SFCONS IN DEGOUTSF, 360 CDUO)
3	OCT	00714	#	DP DEGREES (90) (POINT BETWN BITS 7-8)
4	OCT	31463		
5	OCT	13412	#	DP DEGREES (360)
6	OCT	07534		
7	OCT	05605	#	DEGREES (180)
8	OCT	03656		
9	OCT	00001	#	WEIGHT2
10	OCT	16170		
11	OCT	00441	#	POSITION5
12	OCT	34306		
13	OCT	07176	#	POSITION4 (POINT BETWN BITS 7-8)
14	OCT	21603		
15	OCT	15340	#	VELOCITY2
16	OCT	15340		
17	OCT	01031	#	VELOCITY3 (POINT BETWN BITS 7-8)
18	OCT	21032		
19	OCT	34631	#	ELEVATION DETREES
20	OCT	23146		
21	OCT	00636	#	RENDEZVOUS RADAR RANGE
22	OCT	14552		
23	OCT	74552	#	RENDEZVOUS RADAR RANGE RATE
24	OCT	70307		
25	2DEC	1.079 E-5 B14	#	LANDING RADAR ALTITUDE
26				
27	OCT	14226	#	INITIAL/FINAL ALTITUDE
28	OCT	31757		
29	OCT	02476	#	ALTITUDE RATE
30	OCT	05531		
31	OCT	02727	#	FORWARD/LATERAL VELOCITY
32	OCT	16415		
33	OCT	00007	#	ROT HAND CONT ANGLE RATE
34	OCT	13734		
35	2DEC	-.6440 E-5 B14	#	LANDING RADAR VELX
36				
37	2DEC	1.212 E-5 B14	#	LANDING RADAR VELY
38				
39	2DEC	.8668 E-5 B14	#	LANDING RADAR VELZ
40				
41	OCT	34772	#	POSITION7
42	OCT	07016		
43	OCT	01030	#	TRIM DEGREES2
44	OCT	33675		
45	OCT	01046	#	COMPUTED ALTITUDE
46	OCT	15700		
47	OCT	00321	#	DP DEGREES
48	OCT	26706		
49	2DEC	17.2010499 B-7	#	POSITION 9
50				
51	2DEC	.032808399	#	VELOCITY4
52				
53				
54				
55				
56				
57				
58				
59				
60				

1						1
2	2DEC	.32	#	RADIANS		2
3						3
4	# END OF SFOUTAB					4
5						5
6						6
7						7
8						8
9	IDADDTAB	ECADR	TTOGO	# NN	SF CONSTANT	SF ROUTINE
10		ECADR	VGDISP	# 40	MIN/SEC	M/S
11		ECADR	DVTOTAL	# 40	VEL3	DP3
12		ECADR	DSPTM1	# 40	VEL3	DP3
13		ECADR	DSPTM1 +1	# 41	CDU DEG	CDU
14		ECADR	OCT 0	# 41	ELEV DEG	ARTH
15		OCT	0	# 41	SPARE COMPONENT	
16		ECADR	HAP0	# 42	POS4	DP3
17		ECADR	HPER	# 42	POS4	DP3
18		ECADR	VGDISP	# 42	VEL3	DP3
19		ECADR	LAT	# 43	DPDEG(360)	DP4
20		ECADR	LONG	# 43	DPDEG(360)	DP4
21		ECADR	ALT	# 43	POS4	DP3
22		ECADR	HAP0X	# 44	POS4	DP3
23		ECADR	HPERX	# 44	POS4	DP3
24		ECADR	TFF	# 44	MIN/SEC	M/S
25		ECADR	TRKMKCNT	# 45	WHOLE	ARTH
26		ECADR	TTOGO	# 45	MIN/SEC	M/S
27		ECADR	+MGA	# 45	DPDEG(360)	DP4
28		ECADR	DAPDATR1	# 46	OCTAL ONLY	OCT
29		OCT	0	# 46	SPARE COMPONENT	
30		OCT	0	# 46	SPARE COMPONENT	
31		ECADR	LEMMASS	# 47	WEIGHT2	ARTH1
32		ECADR	CSMMASS	# 47	WEIGHT2	ARTH1
33		OCT	0	# 47	SPARE COMPONENT	
34		ECADR	PITTIME	# 48	TRIM DEG2	ARTH
35		ECADR	ROLLTIME	# 48	TRIM DEG2	ARTH
36		OCT	0	# 48	SPARE COMPONENT	
37		ECADR	R22DISP	# 49	POS4	DP3
38		ECADR	R22DISP +2	# 49	VEL3	DP3
39		ECADR	WHCHREAD	# 49	WHOLE	ARTH
40		OCT	0	# 50	SPARE	
41		OCT	0	# 50	SPARE	
42		OCT	0	# 50	SPARE	
43		ECADR	ALPHASB	# 51	DPDEG(360)	DP4
44		ECADR	BETASB	# 51	DPDEG(360)	DP4
45		OCT	0	# 51	SPARE COMPONENT	
46		ECADR	ACTCENT	# 52	DPDEG(360)	DP4
47		OCT	00000	# 52	SPARE COMPONENT	
48		OCT	00000	# 52	SPARE COMPONENT	
49		OCT	00000	# 52	SPARE COMPONENT	
50		OCT	00000	# 53	SPARE	
51		OCT	00000	# 53		
52		OCT	00000	# 53		
53		ECADR	RANGE	# 54	POS5	DP1
54						
55						
56						
57						
58						
59						
60						

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	1
78	
79	
80	

76	1
77	

OCT	0	# 87	SPARE COMPONENT	
ECADR	STARAD	# 88	FRAC	FRAC
ECADR	STARAD +2	# 88	FRAC	FRAC
ECADR	STARAD +4	# 88	FRAC	FRAC
ECADR	LANDLAT	# 89	DPDEG(90)	DP3
ECADR	LANDLONG	# 89	DPDEG(90)	DP3
ECADR	LANDALT	# 89	POS5	DP1
ECADR	RANGE	# 90	POS5	DP1
ECADR	RRATE	# 90	VEL3	DP3
ECADR	RTHETA	# 90	DPDEG(360)	DP4
ECADR	P21ALT	# 91	POS4	DP3
ECADR	P21VEL	# 91	VEL2	DP4
ECADR	P21GAM	# 91	DPDEG(360)	DP4
OCT	00000	# 92	SPARE	
OCT	00000	# 92		
OCT	00000	# 92		
ECADR	OGC	# 93	DPDEG(90)	DP3
ECADR	OGC +2	# 93	DPDEG(90)	DP3
ECADR	OGC +4	# 93	DPDEG(90)	DP3
OCT	00000	# 94	SPARE	
OCT	00000	# 94		
OCT	00000	# 94		
OCT	0	# 95	SPARE	
OCT	0	# 95	SPARE	
OCT	0	# 95	SPARE	
OCT	0	# 96	SPARE	
OCT	0	# 96	SPARE	
OCT	0	# 96	SPARE	
ECADR	DSPTM1	# 97	WHOLE	ARTH
ECADR	DSPTM1 +1	# 97	WHOLE	ARTH
ECADR	DSPTM1 +2	# 97	WHOLE	ARTH
ECADR	DSPTM2	# 98	WHOLE	ARTH
ECADR	DSPTM2 +1	# 98	FRAC	FRAC
ECADR	DSPTM2 +2	# 98	WHOLE	ARTH
ECADR	WWPOS	# 99	POS9	DP3
ECADR	WWVEL	# 99	VEL4	DP2
ECADR	WWBIAS	# 99	RADIANS	DP4
# END OF IDADDTAB				
# NN SF ROUTINES				
RUTMXTAB	OCT 16351	# 40	M/S, DP3, DP3	
	OCT 00142	# 41	CDU, ARTH	
	OCT 16347	# 42	DP3, DP3, DP3	
	OCT 16512	# 43	DP4, DP4, DP3	
	OCT 22347	# 44	DP3, DP3, M/S	
	OCT 24443	# 45	ARTH, M/S, DP4	
	OCT 00000	# 46	OCT	
	OCT 00553	# 47	ARITH1, ARITH1	

OCT	00143	# 48	ARTH, ARTH
OCT	06347	# 49	DP3, DP3, ARTH
OCT	0	# 50	SPARE
OCT	00512	# 51	DP4, DP4
OCT	00012	# 52	DP4
OCT	00000	# 53	SPARE
OCT	24344	# 54	DP1, DP3, DP4
OCT	24503	# 55	ARTH, DP4, DP4
OCT	00512	# 56	DP4, DP4
OCT	00007	# 57	DP3
OCT	16347	# 58	DP3, DP3, DP3
OCT	16347	# 59	DP3, DP3, DP3
OCT	10347	# 60	DP3, DP3, DP1
OCT	24451	# 61	M/S, M/S, DP4
OCT	16447	# 62	DP3, M/S, DP3
OCT	10347	# 63	DP3, DP3, DP1
OCT	10354	# 64	2INT, DP3, DP1
OCT	20410	# 65	HMS, HMS, HMS
OCT	00304	# 66	DP1, LRPOS
OCT	10204	# 67	DP1, DP1, DP1
OCT	10452	# 68	DP4, M/S, DP1
OCT	00000	# 69	SPARE
OCT	0	# 70	OCT, OCT, OCT
OCT	0	# 71	OCT, OCT, OCT
OCT	00115	# 72	360-CDU, CDU
OCT	00115	# 73	360-CDU, CDU
OCT	24511	# 74	M/S, DP4, DP4
OCT	22447	# 75	DP3, M/S, M/S
OCT	16347	# 76	DP3, DP3, DP3
OCT	00351	# 77	M/S, DP3
OCT	00204	# 78	DP1, DP1
OCT	06102	# 79	CDU, CDU, ARTH
OCT	00503	# 80	ARTH, DP4
OCT	16347	# 81	DP3, DP3, DP3
OCT	16347	# 82	DP3, DP3, DP3
OCT	16347	# 83	DP3, DP3, DP3
OCT	16347	# 84	DP3, DP3, DP3
OCT	16347	# 85	DP3, DP3, DP3
OCT	16347	# 86	DP3, DP3, DP3
OCT	00102	# 87	CDU, CDU
OCT	02041	# 88	FRAC FOR EACH
OCT	10347	# 89	DP3, DP3, DP1
OCT	24344	# 90	DP1, DP3, DP4
OCT	24507	# 91	DP3, DP4, DP4
OCT	00000	# 92	SPARE
OCT	16347	# 93	DP3, DP3, DP3
OCT	00000	# 94	SPARE
OCT	0	# 95	SPARE
OCT	0	# 96	SPARE
OCT	06143	# 97	ARTH, ARTH, ARTH



PINBALL_NOUN_TABLES

PAGE 319

1412THE

1					1
2		OCT	06043	# 98	ARTH, FRAC, ARTH
3		OCT	24247	# 99	DP3, DP2, DP4
4					5
5	# END OF RUTMXTAB				6
6					7
7	SBANK= LOWSUPER				8
8					9
9					10
10					11
11					12
12					13
13					14
14					15
15					16
16					17
17					18
18					19
19					20
20					21
21					22
22					23
23					24
24					25
25					26
26					27
27					28
28					29
29					30
30					31
31					32
32					33
33					34
34					35
35					36
36					37
37					38
38					39
39					40
40					41
41					42
42					43
43					44
44					45
45					46
46					47
47					48
48					49
49					50
50					51
51					52
52					53
53					54
54					55
55					56
56					57
57					58
58					59
59					60
60					61
					62
					63
					64
					65
					66
					67
					68
					69
					70
					71
					72
					73
					74
					75
					76
					77
					78
					79
					80

```

1
2      BANK      23
3      SETLOC    LEMGEOM
4      BANK
5
6      SBANK=    LOWSUPER
7      EBANK=    XSM
8
9      # THESE TWO ROUTINES COMPUTE THE ACTUAL STATE VECTOR FOR LM,CSM BY ADDING
10     # THE CONIC R,V AND THE DEVIATIONS R,V. THE STATE VECTORS ARE CONVERTED TO
11     # METERS B-29 AND METERS/CSEC B-7 AND STORED APPROPRIATELY IN RN,VN OR
12     # R-OTHER , V-OTHER FOR DOWNLINK. THE ROUTINES NAMES ARE SWITCHED IN THE
13     # OTHER VEHICLES COMPUTER.
14     #
15     # INPUT
16     #   STATE VECTOR IN TEMPORARY STORAGE AREA
17     #   IF STATE VECTOR IS SCALED POS B27 AND VEL B5
18     #     SET X2 TO +2
19     #   IF STATE VECTOR IS SCALED POS B29 AND VEL B7
20     #     SET X2 TO 0
21     #
22     # OUTPUT
23     #   R(T) IN RN, V(T) IN VN, T IN PIPTIME
24     # OR
25     #   R(T) IN R-OTHER, V(T) IN V-OTHER   (T IS DEFINED BY T-OTHER)
26
27     COUNT*    $$/GEOM
28     SVDWN2    BOF      RVQ          # SW=1=AVETOMID DOING W-MATRIX INTEG.
29              AVEMIDSW
30              +1
31     VLOAD     VSL*
32              TDELTA V
33              0 -7,2
34     VAD       VSL*
35              RCV
36              0,2
37     STOVL     RN
38              TNUV
39     VSL*      VAD
40              0 -4,2
41              VCV
42     VSL*
43              0,2
44     STODL     VN
45              TET
46     STORE     PIPTIME
47     RVQ
48
49
50
51
52
53
54
55
56
57
58
59
60

```



1				1
2	SVDWN1	VLOAD	VSL*	2
3			TDELTAV	3
4			0 -7,2	4
5		VAD	VSL*	5
6			RCV	6
7			0,2	7
8		STOVL	R-OTHER	8
9			TNUV	9
10		VSL*	VAD	10
11			0 -4,2	11
12			VCV	12
13		VSL*		13
14			0,2	14
15		STORE	V-OTHER	15
16		RVQ		16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

```
# THE FOLLOWING ROUTINE TAKES A HALF UNIT TARGET VECTOR REFERRED TO NAV BASE COORDINATES AND FINDS BOTH
# GIMBAL ORIENTATIONS AT WHICH THE RR MIGHT SIGHT THE TARGET. THE GIMBAL ANGLES CORRESPONDING TO THE PRESENT MODE
# ARE LEFT IN MODEA AND THOSE WHICH WOULD BE USED AFTER A REMODE IN MODEB. THIS ROUTINE ASSUMES MODE 1 IS TRUNNION
# ANGLE LESS THAN 90 DEGS IN ABS VALUE WITH ARBITRARY SHAFT, WITH A CORRESPONDING DEFINITION FOR MODE 2. MODE
# SELECTION AND LIMIT CHECKING ARE DONE ELSEWHERE.
```

```
# THE MODE 1 CONFIGURATION IS CALCULATED FROM THE VECTOR AND THEN MODE 2 IS FOUND USING THE RELATIONS
```

```
# S(2) = 180 + S(1)
# T(2) = 180 - T(1)
```

```
# THE VECTOR ARRIVES IN MPAC WHERE TRG*SMNB OR *SMNB* WILL HAVE LEFT IT.
```

```
RRANGLES      STORE  32D
                DLOAD  DCOMP      # SINCE WE WILL FIND THE MODE 1 SHAFT
                34D      # ANGLE LATER, WE CAN FIND THE MODE 1
                SETPD  ASIN      # TRUNNION BY SIMPLY TAKING THE ARCSIN OF
                0        # THE Y COMPONENT, THE ASIN GIVIN AN
                PUSH   BDSU      # ANSWER WHOSE ABS VAL IS LESS THAN 90 DEG
                LODPHALF
                STODL   4        # MODE 2 TRUNNION TO 4.
                LO6ZEROS
                STOVL  34D      # UNIT THE PROJECTION OF THE VECTOR
                32D      # IN THE X-Z PLANE
                UNIT   BOVB      # IF OVERFLOW, TARGET VECTOR IS ALONG Y
                LUNDESCH      # CALL FOR MANEUVER UNLESS ON LUNAR SURF
                STODL  32D      # PROJECTION VECTOR.
                32D
                SR1     STQ
                32D
                STODL  SINTH      # USE ARCTRIG SINCE SHAFT COULD BE ARB.
                36D
                SR1
                STCALL  COSTH
                ARCTRIG
```



1					1
2		PUSH	DAD	# MODE 1 SHAFT TO 2.	2
3			LODPHALF		3
4		STOVL	6		4
5			4		5
6		RTB		# FIND MODE 2 CDU ANGLES.	6
7			2V1ST02S		7
8		STOVL	MODEB		8
9			0		9
10		RTB		# MODE 1 ANGLES TO MODE A.	10
11			2V1ST02S		11
12		STORE	MODEA		12
13		EXIT			13
14					14
15		CS	RADMODES	# SWAP MODEA AND MODEB IF RR IN MODE 2.	15
16		MASK	ANTENBIT		16
17		CCS	A		17
18		TCF	+4		18
19					19
20		DXCH	MODEA		20
21		DXCH	MODEB		21
22		DXCH	MODEA		22
23					23
24		TC	INTPRET		24
25		GOTO			25
26			S2		26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

GIVEN RR TRUNNION AND SHAFT (T,S) IN TANGNB,+1,FIND THE ASSOCIATED
LINE OF SIGHT IN NAV BASE AXES. THE HALF UNIT VECTOR, .5(SIN(S)COS(T),
-SIN(T),COS(S)COS(T)) IS LEFT IN MPAC AND 32D.

SETLOC INFLIGHT
BANK

COUNT* \$\$/GEOM

RRNB

SLOAD

RTB

TANGNB

SETPD

CDULOGIC

PUSH

TRUNNION ANGLE TO 0

0

SIN

DCOMP

STODL

34D

Y COMPONENT

COS

SLOAD

PUSH

.5 COS(T) TO 0

RTB

TANGNB +1

CDULOGIC

RRNB1

PUSH

COS

SHAFT ANGLE TO 2

DMP

SL1

0

STODL

36D

Z COMPONENT

SIN

DMP

SL1

STOVL

32D

32D

RVQ

THIS ENTRY TO RRNB REQUIRES THE TRUNNION AND SHAFT ANGLES IN MPAC AND MPAC +1 RESPECTIVELY

RRNBMPAC

STODL

20D

SAVE SHAFT CDU IN 21.

MPAC

SET MODE TO DP. (THE PRECEEDING STORE

MAY BE DP. TP OR VECTOR.)

RTB

SETPD

CDULOGIC

0

PUSH

SIN

TRUNNION ANGLE TO 0

DCOMP

STODL

34D

Y COMPONENT

COS

PUSH

.5COS(T) TO 0

SLOAD

RTB

PICK UP CDU'S.

21D

CDULOGIC

GOTO

RRNB1


```
1 BANK 7
2 SETLOC IMUCOMP
3 BANK
4
5 EBANK= NBDX
6
7
8 COUNT* $$/ICOMP
9 1/PIPA CAF LGCOMP # SAVE EBANK OF CALLING PROGRAM
10 XCH EBANK
11 TS MODE
12
13 CCS GCOMP SW # BYPASS IF GCOMP SW NEGATIVE
14 TCF +3
15 TCF +2
16 TCF IRIG1 # RETURN
17
18 1/PIPA1 CAF FOUR # PIPAZ, PIPAY, PIPAX
19 TS BUF +2
20
21 INDEX BUF +2
22 CA PIPASCF # (P.P.M.) X 2(-9)
23 EXTEND
24 INDEX BUF +2
25 MP DELVX # (PP) X 2(+14) NOW (PIPA PULSES) X 2(+5)
26 TS Q # SAVE MAJOR PART
27
28 CA L # MINOR PART
29 EXTEND
30 MP BIT6 # SCALE 2(+9) SHIFT RIGHT 9
31 INDEX BUF +2
32 TS DELVX +1 # FRACTIONAL PIPA PULSES SCALED 2(+14)
33
34 CA Q # MAJOR PART
35 EXTEND
36 MP BIT6 # SCALE 2(+9) SHIFT RIGHT 9
37 INDEX BUF +2
38 DAS DELVX # (PIPAI) + (PIPAI)(SFE)
39
40 INDEX BUF +2
41 CS PIPABIAS # (PIPA PULSES)/(CS) X 2(-5) *
42 EXTEND
43 MP 1/PIPADT # (CS) X 2(+8) NOW (PIPA PULSES) X 2(+3)*
44 EXTEND
45 MP BIT4 # SCALE 2(+11) SHIFT RIGHT 11 *
46 INDEX BUF +2
47 DAS DELVX # (PIPAI) + (PIPAI)(SFE) - (BIAS)(DELTAT)
48
49 CCS BUF +2 # PIPAZ, PIPAY, PIPAX
50 AD NEG1
51 TCF 1/PIPA1 +1
```




IMU_COMPENSATION_PACKAGE

PAGE 327



1412THE

1			1
2			2
3			3
4			4
5			5
6			6
7			7
8			8
9			9
10			10
11			11
12			12
13			13
14			14
15			15
16			16
17			17
18			18
19			19
20			20
21			21
22			22
23			23
24			24
25			25
26			26
27			27
28			28
29			29
30			30
31			31
32			32
33			33
34			34
35			35
36			36
37			37
38			38
39			39
40			40
41			41
42			42
43			43
44			44
45			45
46			46
47			47
48			48
49			49
50			50
51			51
52			52
53			53
54			54
55			55
56			56
57			57
58			58
59			59
60			60

NOOP

LESS THAN ZERO IMPOSSIBLE

IRIGCOMP	TS TS	GCOMP SW BUF	# INDICATE COMMANDS 2 PULSES OR LESS. # INDEX COUNTER . IRIGX, IRIGY, IRIGZ.
	TC	IRIGX	# COMPENSATE ACCELERATION TERMS
	CS TC	NBDX DRIFTSUB	# (GYRO PULSES)/(CS) X 2(-5) # -(NBOX)(DELTAT) (GYRO PULSES) X 2(+14)
	TC	IRIGY	# COMPENSATE ACCELERATION TERMS
	CS TC	NBDY DRIFTSUB	# (GYRO PULSES)/(CS) X 2(-5) # -(NBDY)(DELTAT) (GYRO PULSES) X 2(+14)
	TC	IRIGZ	# COMPENSATE ACCELERATION TERMS
	CA TC	NBDZ DRIFTSUB	# (GYRO PULSES)/(CS) X 2(-5) # +(NBDZ)(DELTAT) (GYRO PULSES) X 2(+14)
	CCS TCF	GCOMP SW +2	# ARE GYRO COMMANDS GREATER THAN 2 PULSES # YES SEND OUT GYRO TORQUING COMMANDS.
	TCF	IRIG1	# NO RETURN
	CA TC	PRI021 NOVAC	# PRI0 GREATER THAN SERVICER # SEND OUT GYRO TORQUING COMMANDS.
	EBANK= 2CADR	NBDX 1/GYRO	
IRIG1	RELINT CA	MODE	# RESTORE CALLERS EBANK
	TS	EBANK	
	TCF	SWRETURN	

IRIGX

EXTEND

QXCH MPAC +2

SAVE Q

EXTEND

DCS DELVX

(PIPA PULSES) X 2(+14)

DXCH MPAC

CA ADIAX

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

TC GCOMPSUB

-(ADIAX)(PIPAX) (GYRO PULSES) X 2(+14)

EXTEND

DCS DELVY

#

(PIPA PULSES) X 2(+14)

DXCH MPAC

#

CS ADSRAX

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

TC GCOMPSUB

+(ADSRAX)(PIPAY) (GYRO PULSES) X 2(+14)

#

EXTEND

#

DCS DELVZ

(PIPA PULSES) X 2(+14)

#

DXCH MPAC

#

CA ADOAX

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

#

TC GCOMPSUB

-(ADOAX)(PIPAZ) (GYRO PULSES) X 2(+14)

TC MPAC +2

IRIGY

EXTEND

QXCH MPAC +2

SAVE Q

EXTEND

DCS DELVY

(PIPA PULSES) X 2(+14)

DXCH MPAC

CA ADIAY

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

TC GCOMPSUB

-(ADIAY)(PIPAY) (GYRO PULSES) X 2(+14)

EXTEND

DCS DELVZ

(PIPA PULSES) X 2(+14)

DXCH MPAC

CS ADSRAY

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

TC GCOMPSUB

+(ADSRAY)(PIPAZ) (GYRO PULSES) X 2(+14)

#

EXTEND

#

DCS DELVX

(PIPA PULSES) X 2(+14)

#

DXCH MPAC

#

CA ADOAY

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

#

TC GCOMPSUB

-(ADOAY)(PIPAX) (GYRO PULSES) X 2(+14)

TC MPAC +2

IRIGZ

EXTEND

QXCH MPAC +2

SAVE Q

EXTEND

DCS DELVY

(PIPA PULSES) X 2(+14)

DXCH MPAC

CA ADSRAZ

(GYRO PULSES)/(PIPA PULSE) X 2(-6) *

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	

1						1
2	GCOMPSUB	XCH	MPAC	# ADIA OR ADSRA COEFFICIENT ARRIVES IN A		2
3		EXTEND		# C(MPAC) = (PIPA PULSES) X 2(+14)		3
4		MP	MPAC	# (GYRO PULSES)/(PIPA PULSE) X 2(-6)	*	4
5		DXCH	VBUF	# NOW = (GYRO PULSES) X 2(+8)	*	5
6						6
7		CA	MPAC +1	# MINOR PART OF PIPA PULSES		7
8		EXTEND				8
9		MP	MPAC	# ADIA OR ADSRA		9
10		TS	L			10
11		CAF	ZERO			11
12		DAS	VBUF	# NOW = (GYRO PULSES) X 2(+8)	*	12
13						13
14		CA	VBUF	# PARTIAL RESULT - MAJOR		14
15		EXTEND				15
16		MP	BIT9	# SCALE 2(+6) SHIFT RIGHT	*	16
17		INDEX	BUF	# RESULT = (GYRO PULSES) X 2(+14)		17
18		DAS	GCOMP	# HI(ADIA)(PIPAI) OR HI(ADSRA)(PIPAI)		18
19						19
20		CA	VBUF +1	# PARTIAL RESULT - MINOR		20
21		EXTEND				21
22		MP	BIT9	# SCALE 2(+6) SHIFT RIGHT 6	*	22
23		TS	L			23
24		CAF	ZERO			24
25		INDEX	BUF	# RESULT = (GYRO PULSES) X 2(+14)		25
26		DAS	GCOMP	# (ADIA)(PIPAI) OR (ADSRA)(PIPAI)		26
27						27
28		TC	Q			28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

DRIFTSUB

EXTEND
QXCH

BUF +1

EXTEND

MP 1/PIPADT

C(A) = NBD (GYRO PULSES)/(CS) X 2(-5)
(CS) X 2(+8) NOW (GYRO PULSES) X 2(+3)

LXCH

MPAC +1

SAVE FOR FRACTIONAL COMPENSATION

EXTEND

MP

BIT4

SCALE 2(+11) SHIFT RIGHT 11

INDEX

DAS

BUF

GCOMP

HI(NBD)(DELTAT) (GYRO PULSES) X 2(+14)

CA

MPAC +1

NOW MINOR PART

EXTEND

MP

BIT4

SCALE 2(+11) SHIFT RIGHT 11

TS

L

CAF

ZERO

INDEX

BUF

ADD IN FRACTIONAL COMPENSATION

DAS

GCOMP

(NBD)(DELTAT) (GYRO PULSES) X 2(+14)

DRFTSUB2

CAF

TWO

PIPAX, PIPAY, PIPAZ

AD

BUF

XCH

BUF

INDEX

A

CCS

GCOMP

ARE GYRO COMMANDS 1 PULSE OR GREATER

TCF

+2

YES

TC

BUF +1

NO

MASK

COMPCHK

DEC -1

CCS

A

ARE GYRO COMMANDS GREATER THAN 2 PULSES

TS

GCOMP SW

YES - SET GCOMP SW POSITIVE

TC

BUF +1

NO

1/GYRO

CAF
TSFOUR
BUF

PIPAZ, PIPAY, PIPAX

INDEX
CABUF
GCOMP +1# SCALE GYRO COMMANDS FOR IMUPULSE
FRACTIONAL PULSESEXTEND
MP
INDEXBIT8
BUF

SHIFT RIGHT 7

TS

GCOMP +1

FRACTIONAL PULSES SCALED

CAF

ZERO

SET GCOMP = 0 FOR DAS INSTRUCTION

INDEX
XCH
EXTENDBUF
GCOMP

GYRO PULSES

MP
INDEX
DASBIT8
BUF
GCOMP

SHIFT RIGHT 7

ADD THESE TO FRACTIONAL PULSES ABOVE

CCS
ADBUF
NEG1

PIPAZ, PIPAY, PIPAX

LGCOMP

TCF
ECADR1/GYRO +1
GCOMP

LESS THAN ZERO IMPOSSIBLE

CAF
TCLGCOMP
BANKCALL

CADR

IMUPULSE

CALL GYRO TORQUING ROUTINE

TC
CADRBANKCALL
IMUSTALL# WAIT FOR PULSES TO GET OUT
TEMPORARY

TCF

ENDOFJOB

GCOMP1

CAF
TSFOUR
BUF

PIPAZ, PIPAY, PIPAX

INDEX
CABUF
GCOMP +1

RESCALE

EXTEND
MP
INDEXBIT8
BUF

SHIFT MINOR PART LEFT 7 - MAJOR PART = 0

LXCH

GCOMP +1

BITS 8-14 OF MINOR PART WERE = 0

CCS
ADBUF
NEG1

PIPAZ, PIPAY, PIPAX

TCF

GCOMP1 +1

COMPCHK

DEC

-1

LESS THAN ZERO IMPOSSIBLE

TCF

ENDOFJOB

NBDONLY	CCS TCF	GCOMP SW +3	# BYPASS IF GCOMP SW NEGATIVE
	TCF TCF	+2 ENDOFJOB	
	INHINT CCS TCF	FLAGWRD2 ENDOFJOB	# PREREAD T3RUPT MAY COINCIDE
	TCF TCF	ENDOFJOB +1	
	CA MASK TS	FLAGWRD8 BIT8 TEM1	# IF SURFACE FLAG IS SET, SET TEM1 # POSITIVE SO THAT THE ACCELERATION TERMS # WILL BE COMPENSATED.
	EXTEND BZF	+3	# ARE WE ON THE SURFACE
	TC CADR	IBNKCALL PIPASR +3	# ON THE SURFACE # READ PIPAS, BUT DO NOT SCALE THEM
	CA XCH RELINT	TIME1 1/PIPADT	# (CS) X 2(+14) # PREVIOUS TIME
NBD2	COM AD AD	1/PIPADT HALF	# PRESENT TIME - PREVIOUS TIME # CORRECT FOR POSSIBLE TIME1 TICK
	AD XCH XCH	HALF L L	# IF TIME1 DID NOT TICK, REMOVE RESULTING # OVERFLOW.
NBD3	EXTEND MP	BIT10	# C(A) = DELTAT (CS) X 2(+14) # SHIFT RIGHT 5
	DXCH	VBUF +2	
	CA TS TS	ZERO GCOMP SW BUF	# INDICATE COMMANDS 2 PULSES OR LESS. # INDEX X, Y, Z.
	CCS TC	TEM1 IRIGX	# IF SURFACE FLAG IS SET, # COMPENSATE ACCELERATION TERMS.
	EXTEND DCA DXCH	VBUF +2 MPAC	# DELTAT NOW SCALED (CS) X 2(+19)
	CS TC	NBDX FBIASSUB	# (GYRO PULSES)/(CS) X 2(-5) # -(NBDX)(DELTAT) (GYRO PULSES) X 2(+14)
	CCS TC	TEM1 IRIGY	# IF SURFACE FLAG IS SET, # COMPENSATE ACCELERATION TERMS.

THE	2
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
80	

FBIASSUB	XCH TS	Q BUF +1	
	CA EXTEND	Q	# NBD SCALED (GYRO PULSES)/(CS) X 2(-5)
	MP INDEX DAS	MPAC BUF GCOMP	# DELTAT SCALED (CS) X 2(+19) # HI(NBD)(DELTAT) (GYRO PULSES) X 2(+14)
	CA EXTEND	Q	# NOW FRACTIONAL PART
	MP TS CAF	MPAC +1 L ZERO	
	INDEX DAS	BUF GCOMP	# (NBD)(DELTAT) (GYRO PULSES) X 2(+14)
	TCF	DRFTSUB2	# CHECK MAGNITUDE OF COMPENSATION
LASTBIAS	TC CADR	BANKCALL PIPUSE1	
	CCS	GCOMP SW	
	TCF	+3	
	TCF	+2	
	TCF	ENDOFJOB	
	CA MASK TS	FLAGWRD8 SURFFBIT TEM1	# IF SURFACE FLAG IS SET, SET TEM1 # POSITIVE SO THAT THE ACCELERATION TERMS # WILL BE COMPENSATED.
	CAF	PRI031	# 2 SECONDS SCALED (CS) X 2(+8)
	XCH COM AD	1/PIPADT PIPTIME +1	
	TCF	NBD2	
GCOMPZER	CAF XCH TS	LGCOMP EBANK MODE	# ROUTINE TO ZERO GCOMP BEFORE FIRST # CALL TO 1/PIPA
	CAF TS TS	ZERO GCOMP SW GCOMP	
	TS	GCOMP +1	
	TS	GCOMP +2	
	TS	GCOMP +3	
	TS	GCOMP +4	



IMU_COMPENSATION_PACKAGE

PAGE 337



1412THE

1					1
2					2
3					3
4					4
5					5
6					6
7					7
8					8
9					9
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

```
1  # SUBROUTINE NAME:      V89CALL
2
3  # MOD NO:      0          DATE:      9 JAN 1968
4  # MOD BY:      DIGITAL DEVEL GROUP  LOG SECTION:  R63
5  #
6  # FUNCTIONAL DESCRIPTION:
7  #
8  # CALLED BY VERB 89 ENTER DURING P00.  PRIO 10 USED.  CALCULATES AND
9  # DISPLAYS FINAL FDAI BALL ANGLES TO POINT LM +X OR +Z AXIS AT CSM.
10 #
11 # 1. KEY IN V 89 E ONLY IF IN PROG 00.  IF NOT IN P00, OPERATOR ERROR AND
12 # EXIT R63, OTHERWISE CONTINUE.
13 #
14 # 2. IF IN P00, DO IMU STATUS CHECK ROUTINE (R02BOTH).  IF IMU ON AND ITS
15 # ORIENTATION KNOWN TO LGC, CONTINUE.
16 #
17 # 3. FLASH DISPLAY V 04 N 06.  R2 INDICATES WHICH SPACECRAFT AXIS IS TO
18 # BE POINTED AT CSM.  INITIAL CHOICE IS PREFERRED (+Z) AXIS (R2=1).
19 # ASTRONAUT CAN CHANGE TO (+X) AXIS (R2 NOT = 1) BY V 22 E 2 E.  CONTINUE
20 # AFTER KEYING IN PROCEED.
21 #
22 # 4. BOTH VEHICLE STATE VECTORS UPDATED BY CONIC EQS.
23 #
24 # 5. HALF MAGNITUDE UNIT LOS VECTOR (IN STABLE MEMBER COORDINATES) AND
25 # HALF MAGNITUDE UNIT SPACECRAFT AXIS VECTOR (IN BODY COORDINATES)
26 # PREPARED FOR VECPOINT.
27 #
28 # 6. GIMBAL ANGLES FROM VECPOINT TRANSFORMED INTO FDAI BALL ANGLES BY
29 # BALLANGS.  FLASH DISPLAY V 06 N 18 AND AWAIT RESPONSE.
30 #
31 # 7. RECYCLE - RETURN TO STEP 4.
32 #   TERMINATE - EXIT R63.
33 #   PROCEED - RESET 3AXISFLG AND CALL R60LEM FOR ATTITUDE MANEUVER.
34 #
35 # CALLING SEQUENCE:      V 89 E.
36 #
37 # SUBROUTINES CALLED:    CHKPOOH, R02BOTH, GOXDSPF, CSMCONIC, LEMCONIC,
38 #                        VECPOINT, BALLANGS, R60LEM.
39 #
40 # NORMAL EXIT MODES:     TC ENDEXT
41 #
42 # ALARMS:      1. OPERATOR ERROR IF NOT IN P00.
43 #              2. PROGRAM ALARM IF IMU IS OFF.
44 #              3. PROGRAM ALARM IF IMU ORIENTATION IS UNKNOWN.
45 #
46 # OUTPUT:      NONE
47 #
48 # ERASABLE INITIALIZATION REQUIRED:  NONE
49 #
50 # DEBRIS:      OPTION1, +1, TDEC1, PDINTVSM, SCAXIS, CPHI, CTHETA, CPSI,
```

```
1  #
2  3AXISFLG.
3
4  EBANK= RONE
5  BANK 32
6  SETLOC BAWLANGS
7  BANK
8
9  COUNT* $$/R63
10 V89CALL TC BANKCALL # IMU STATUS CHECK. RETURNS IF ORIENTATION
11 CADR R02BOTH # KNOWN. ALARMS IF NOT.
12 CAF THREE # ALLOW ASTRONAUT TO SELECT DESIRED
13 TS OPTIONX # TRACKING ATTITUDE AXIS.
14 CAF ONE
15 TS OPTIONX +1
16 CAF VB04N12 # V 04 N 12
17 TC BANKCALL
18 CADR GOFLASH
19 TC ENDEXT # TERMINATE
20 TC +2 # PROCEED
21 TC -5 # DATA IN. OPTION1+1 = 1 FOR Z AXIS
22 V89RECL TC INTPRET # = 2 FOR X AXIS
23 RTB DAD
24 LOADTIME # READ PRESENT TIME
25 DP1MIN
26 STORE TSTART82 # SAVE TIME FOR LEMCONIC CALL
27 STCALL TDEC1 # STORE TIME FOR CSMCONIC CALL
28 CSMCONIC # CSM STATE VECTOR UPDATE
29 VLOAD # CSMCONIC LEFT R VECTOR IN RATT
30 RATT
31 STODL RONE # SAVE FOR LINE OF SIGHT (LOS) COMPUTATION
32 TSTART82
33 STCALL TDEC1 # STORE TIME FOR LEMCONIC CALL
34 LEMCONIC # LEM STATE VECTOR UPDATE
35 VLOAD VSU # CSM POSITION - LEM POSITION = LOS
36 RONE # LOS VECTOR LEFT IN MPAC
37 RATT
38 MXV RTB # (REFSMAT X LOS). TRANSFORMS LOS FROM
39 REFSMMAT # REFERENCE COORD TO STAB MEMB COORD.
40 NORMUNIT
41 STORE POINTVSM # STORE LOS FOR VECPOINT CALCULATION
42 EXIT
43 CS OPTIONX +1 # 1 FOR Z AXIS. 2 FOR X AXIS.
44 AD ONE
45 EXTEND
46 ALINEX BZF ALINEZ
47 TC INTPRET # X AXIS ALIGNMENT
48 VLOAD
49 UNITX # READ (.5, 0, 0)
```

1					1
2	V89CALL1	STCALL	SCAXIS	# STORE SELECTED ALIGNMENT AXIS	2
3			VECPOINT	# PUTS DESIRED GIM ANG (OG,IG,MG) IN TMPAC	3
4		STORE	CPHI	# STORE GIMBAL ANGLES FOR BALLANGS CALL.	4
5		EXIT			5
6		TC	BANKCALL		6
7		CADR	BALLANGS	# PUTS DESIRED BALL ANGLE IN FDAIX,Y,Z	7
8		CAF	VB06N18	# V 06 N 18	8
9		TC	BANKCALL	# NOUN 18 REFERS TO FDAIX,Y,Z	9
10		CADR	GOFLASH		10
11		TC	ENDEXT	# TERMINATE	11
12		TC	+2	# PROCEED	12
13		TC	V89RECL	# RECYCLE	13
14		TC	DOWNFLAG	# RESET 3 AXIS FLAG	14
15		ADRES	3AXISFLG	# RESET BIT6 FLAG WORD 5	15
16		TC	BANKCALL	# PERFORMS LEM MANEUVER TO ALIGN SELECTED	16
17		CADR	R60LEM	# SPACECRAFT AXIS TO CSM.	17
18		TCF	ENDEXT	# TERMINATE R63	18
19	ALINEZ	TC	INTPRET	# Z AXIS ALIGNMENT	19
20		VLOAD	GOTO		20
21			UNITZ	# READ (0, 0, .5)	21
22			V89CALL1		22
23					23
24					24
25	VB04N12	VN	412		25
26	VB06N18	VN	0618		26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60



R63

PAGE 341

1				1
2	DP1MIN	2DEC	6000	2
3				3
4				4
5				5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1412THE

BLOCK 2 LGC ATTITUDE MANEUVER ROUTINE -- KALCMANU

#

MOD 2 DATE 5/1/67 BY DON KEENE

#

PROGRAM DESCRIPTION

#

KALCMANU IS A ROUTINE WHICH GENERATES COMMANDS FOR THE LM DAP TO CHANGE THE ATTITUDE OF THE SPACECRAFT
DURING FREE FALL. IT IS DESIGNED TO MANEUVER THE SPACECRAFT FROM ITS INITIAL ORIENTATION TO SOME DESIRED

ORIENTATION SPECIFIED BY THE PROGRAM WHICH CALLS KALCMANU, AVOIDING GIMBAL LOCK IN THE PROCESS. IN THE
MOD 2 VERSION, THIS DESIRED ATTITUDE IS SPECIFIED BY A SET OF OF THREE COMMANDED CDU ANGLES STORES AS 2'S COMPLEMENT
SINGLE PRECISION ANGLES IN THE THREE CONSECUTIVE LOCATIONS, CPHI, CTHETA, CPSI, WHERE

#

CPHI = COMMANDED OUTER GIMBAL ANGLE

CTHETA = COMMANDED INNER GIMBAL ANGLE

CPSI = COMMANDED MIDDLE GIMBAL ANGLE

#

WHEN POINTING A SPACECRAFT AXIS (I.E., X, Y, Z, THE AOT, THRUST AXIS, ETC.) THE SUBROUTINE VECPOINT MAY BE
USED TO GENERATE THIS SET OF DESIRED CDU ANGLES (SEE DESCRIPTION IN R60).

#

WITH THIS INFORMATION KALCMANU DETERMINES THE DIRECTION OF THE SINGLE EQUIVALENT ROTATION (COF ALSO U) AND THE
MAGNITUDE OF THE ROTATION (AM) TO BRING THE S/C FROM ITS INITIAL ORIENTATION TO ITS FINAL ORIENTATION.

THIS DIRECTION REMAINS FIXED BOTH IN INERTIAL COORDINATES AND IN COMMANDED S/C AXES THROUGHOUT THE

#

MANEUVER. ONCE COF AND AM HAVE BEEN DETERMINED, KALCMANU THEN EXAMINES THE MANEUVER TO SEE IF IT WILL BRING

#

THE S/C THROUGH GIMBAL LOCK. IF SO, COF AND AM ARE READJUSTED SO THAT THE S/C WILL JUST SKIM THE GIMBAL
LOCK ZONE AND ALIGN THE X-AXIS. IN GENERAL A FINAL YAW ABOUT X WILL BE NECESSARY TO COMPLETE THE MANEUVER.
NEEDLESS TO SAY, NEITHER THE INITIAL NOR THE FINAL ORIENTATION CAN BE IN GIMBAL LOCK.

#

FOR PROPER ATTITUDE CONTROL THE DIGITAL AUTOPILOT MUST BE GIVEN AN ATTITUDE REFERENCE WHICH IT CAN TRACK.
KALCMANU DOES THIS BY GENERATING A REFERENCE OF DESIRED GIMBAL ANGLES (CDUXD, CDUYD, CDUZD) WHICH ARE UPDATED
EVERY ONE SECOND DURING THE MANEUVER. TO ACHIEVE A SMOOTHER SEQUENCE OF COMMANDS BETWEEN SUCCESSIVE UPDATES,
THE PROGRAM ALSO GENERATES A SET OF INCREMENTAL CDU ANGLES (DELDCDU) TO BE ADDED TO CDU DESIRED BY THE DIGITAL
AUTOPILOT. KALCMANU ALSO CALCULATES THE COMPONENT MANEUVER RATES (OMEGAPD, OMEGAQD, OMEGARD), WHICH CAN

#

BE DETERMINED SIMPLY BY MULTIPLYING COF BY SOME SCALAR (ARATE) CORRESPONDING TO THE DESIRED ROTATIONAL RATE.

#

AUTOMATIC MANEUVERS ARE TIMED WITH THE HELP OF WAITLIST SO THAT AFTER A SPECIFIED INTERVAL THE Y AND Z
DESIRED RATES ARE SET TO ZERO AND THE DESIRED CDU ANGLES (CDUYD, CDUZD) ARE SET EQUAL TO THE FINAL DESIRED CDU
ANGLES (CTHETA, CPSI). IF ANY YAW REMAINS DUE TO GIMBAL LOCK AVOIDANCE, THE FINAL YAW MANEUVER IS
CALCULATED AND THE DESIRED YAW RATE SET TO SOME FIXED VALUE (ROLLRATE = + OR - 2 DEGREES PER SEC).

IN THIS CASE ONLY AN INCREMENTAL CDUX ANGLE (DELFROLL) IS SUPPLIED TO THE DAP. AT THE END OF THE YAW
MANEUVER OR IN THE EVENT THAT THERE WAS NO FINAL YAW, CDUXD IS SET EQUAL TO CPHI AND THE X-AXIS DESIRED
RATE SET TO ZERO. THUS, UPON COMPLETION OF THE MANEUVER THE S/C WILL FINISH UP IN A LIMIT CYCLE ABOUT THE
DESIRED GIMBAL ANGLES.

#

PROGRAM LOGIC FLOW

#

KALCMANU IS CALLED AS A HIGH PRIORITY JOB WITH ENTRY POINTS AT KALCMAN3 AND VECPOINT. IT FIRST PICKS
UP THE CURRENT CDU ANGLES TO BE USED AS THE BASIS FOR ALL COMPUTATIONS INVOLVING THE INITIAL S/C ORIENTATION.


```
1 # IT THEN DETERMINES THE DIRECTION COSINE MATRICES RELATING BOTH THE INITIAL AND FINAL S/C ORIENTATION TO STABLE
2 #
3 # * * *
4 # MEMBER AXES (MIS,MFS). IT ALSO COMPUTES THE MATRIX RELATING FINAL S/C AXES TO INITIAL S/C AXES (MFI). THE
5 # ANGLE OF ROTATION (AM) IS THEN EXTRACTED FROM THIS MATRIX, AND TEST ARE MADE TO DETERMINE IF
6 #
7 # A) AM LESS THAN .25 DEGREES (MINANG)
8 # B) AM GREATER THAN 170 DEGREES (MAXANG)
9 #
10 # IF AM IS LESS THAN .25 DEGREES, NO COMPLICATED AUTOMATIC MANEUVERING IS NECESSARY. THEREFORE, WE CAN SIMPLY
11 # SET CDU DESIRED EQUAL TO THE FINAL CDU DESIRED ANGLES AND TERMINATE THE JOB.
12 #
13 # IF AM IS GREATER THAN .25 DEGREES BUT LESS THAN 170 DEGREES THE AXES OF THE SINGLE EQUIVALENT ROTATION
14 # *
15 # (C̄OF) IS EXTRACTED FROM THE SKEW SYMMETRIC COMPONENTS OF MFI.
16 #
17 # IF AM GREATER THAN 170 DEGREES AN ALTERNATE METHOD EMPLOYING THE SYMMETRIC PART OF MFI (MFISYM) IS USED
18 #
19 # TO DETERMINE C̄OF.
20 #
21 # THE PROGRAM THEN CHECKS TO SEE IF THE MANEUVER AS COMPUTED WILL BRING THE S/C THROUGH GIMBAL LOCK. IF
22 # SO, A NEW MANEUVER IS CALCULATED WHICH WILL JUST SKIM THE GIMBAL LOCK ZONE AND ALIGN THE S/C X-AXIS. THIS
23 # METHOD ASSURES THAT THE ADDITIONAL MANEUVERING TO AVOID GIMBAL LOCK WILL BE KEPT TO A MINIMUM. SINCE A FINAL
24 # P AXIS YAW WILL BE NECESSARY, A SWITCH IS RESET (STATE SWITCH 31) TO ALLOW FOR THE COMPUTATION OF THIS FINAL
25 # YAW.
26 #
27 # AS STATED PREVIOUSLY, KALCMANU GENERATES A SEQUENCE OF DESIRED GIMBAL ANGLES WHICH ARE UPDATED EVERY
28 #
29 # SECOND. THIS IS ACCOMPLISHED BY A SMALL ROTATION OF THE DESIRED S/C FRAME ABOUT THE VECTOR C̄OF. THE NEW
30 # DESIRED REFERENCE MATRIX IS THEN,
31 # * * *
32 # MIS = MIS DEL
33 # N+1 N
34 # *
35 # WHERE DEL IS THE MATRIX CORRESPONDING TO THIS SMALL ROTATION. THE NEW CDU ANGLES CAN THEN BE EXTRACTED
36 # *
37 # FROM MIS.
38 #
39 # AT THE BEGINNING OF THE MANEUVER THE AUTOPILOT DESIRED RATES (OMEGAPD, OMEGAQD, OMEGARD) AND THE
40 # MANEUVER TIMINGS ARE ESTABLISHED. ON THE FIRST PASS AND ON ALL SUBSEQUENT UPDATES THE CDU DESIRED
41 # ANGLES ARE LOADED WITH THE APPROPRIATE VALUES AND THE INCREMENTAL CDU ANGLES ARE COMPUTED. THE AGC CLOCKS
42 # (TIME1 AND TIME2) ARE THEN CHECKED TO SEE IF THE MANEUVER WILL TERMINATE BEFORE THE NEXT UPDATE. IF
43 # NOT, KALCMANU CALLS FOR ANOTHER UPDATE (RUN AS A JOB WITH PRIORITY TBD) IN ONE SECOND. ANY DELAYS IN THIS
44 # CALLING SEQUENCE ARE AUTOMATICALLY COMPENSATED IN CALLING FOR THE NEXT UPDATE.
45 #
46 # IF IT IS FOUND THAT THE MANEUVER IS TO TERMINATE BEFORE THE NEXT UPDATE A ROUTINE IS CALLED (AS A WAIT-
47 # LIST TASK) TO STOP THE MANEUVER AT THE APPROPRIATE TIME AS EXPLAINED ABOVE.
```

#

```
#
# CAF      PRIO XX
```

```
#
# INHINT
```

#	TC	FINDVAC
#	2CADR	KALCMAN3

```
# RELINT
```

```
# THE USER'S PROGRAM MAY EITHER CONTINUE OR WAIT FOR THE TERMINATION OF THE MANEUVER.  IF THE USER WISHES TO
# WAIT. HE MAY PUT HIS JOB TO SLEEP WITH THE FOLLOWING INSTRUCTIONS:
```

#	L	TC	BANKCALL
---	---	----	----------

#	L+1	CADR	ATTSTALL
#	L+2	(BAD RETURN)	

#	L+3	(GOOD RETURN)
#		

```
# UPON COMPLETION OF THE MANEUVER, THE PROGRAM WILL BE AWAKENED AT L+3 IF THE MANEUVER WAS COMPLETED
# SUCCESSFULLY, OR AT L+2 IF THE MANEUVER WAS ABORTED.  THIS ABORT WOULD OCCUR IF THE INITIAL OR FINAL ATTITUDE
# WAS IN GIMBAL LOCK.
```

```
#
# *** NOTA BENE *** IF IT IS ASSUMED THAT THE DESIRED MANEUVERING RATE (0.5, 2, 5, 10 DEG/SEC) HAS BEEN SELECTED BY
# KEYBOARD ENTRY PRIOR TO THE EXECUTION OF KALCMANU.
#
```

```
# IT IS ALSO ASSUMED THAT THE AUTOPILOT IS IN THE AUTO MODE.  IF THE MODE SWITCH IS CHANGED DURING THE
# MANEUVER, KALCMANU WILL TERMINATE VIA GOODEND WITHIN 1 SECOND SO THAT R60 MAY REQUEST A TRIM OF THE S/C ATTITUDE
# SUBROUTINES.
```

```
#
# KALCMANU USES A NUMBER OF INTERPRETIVE SUBROUTINES WHICH MAY BE OF GENERAL INTEREST.  SINCE THESE ROUTINES
# WERE PROGRAMMED EXCLUSIVELY FOR KALCMANU, THEY ARE NOT, AS YET, GENERALLY AVAILABLE FOR USE BY OTHER PROGRAMS.
```

```
#
# MXM3
```

```
# ----
#
# THIS SUBROUTINE MULTIPLIES TWO 3X3 MATRICES AND LEAVES THE RESULT IN THE FIRST 18 LOCATIONS OF THE PUSH
# DOWN LIST. I.E..
```

#		[M M M]						
#		[0 1 2]						
#	*	[]			* *			
#	M =	[M M M]	=		M1 X M2			
#		[3 4 5]						
#		[]						
#		[M M M]						
#		[6 7 8]						

```

1  #
2  # INDEX REGISTER X1 MUST BE LOADED WITH THE COMPLEMENT OF THE STARTING ADDRESS FOR M1, AND X2 MUST BE
3  #
4  # LOADED WITH THE COMPLEMENT OF THE STARTING ADDRESS FOR M2. THE ROUTINE USES THE FIRST 20 LOCATIONS OF THE PUSH
5  # DOWN LIST. THE FIRST ELEMENT OF THE MATRIX APPEARS IN PDO. PUSH UP FOR M .
6  #
7  # TRANSPOS
8  # -----
9  #
10 #
11 # THIS ROUTINE TRANSPOSES A 3X3 MATRIX AND LEAVES THE RESULT IN THE PUSH DOWN LIST, I.E.,
12 #
13 #      *      * T
14 #      M      =  M1
15 #
16 # INDEX REGISTER X1 MUST CONTAIN THE COMPLEMENT OF THE STARTING ADDRESS FOR M1. PUSH UP FOR THE FIRST AND SUB-
17 # SEQUENT COMPONENTS OF M. THIS SUBROUTINE ALSO USES THE FIRST 20 LOCATIONS OF THE PUSH DOWN LIST.
18 #
19 #
20 # CDU TO DCM
21 # -----
22 #
23 # THIS SUBROUTINE CONVERTS THREE CDU ANGLES IN T(MPAC) TO A DIRECTION COSINE MATRIX (SCALED BY 2) RELATING
24 # THE CORRESPONDING S/C ORIENTATIONS TO THE STABLE MEMBER FRAME. THE FORMULAS FOR THIS CONVERSION ARE
25 #
26 #      M      =      COSY COSZ
27 #      0
28 #
29 #      M      =      -COSY SINZ COSX + SINY SINX
30 #      1
31 #
32 #      M      =      COSY SINZ SINX + SINY COSX
33 #      2
34 #
35 #      M      =      SINZ
36 #      3
37 #
38 #      M      =      COSZ COSX
39 #      4
40 #
41 #      M      =      -COSZ SINX
42 #      5
43 #
44 #      M      =      -SINY COSZ
45 #      6
46 #
47 #      M      =      SINY SINZ COSX + COSY SINX
48 #      7

```

ATTITUDE_MANEUVER_ROUTINE

M = -SINY SINZ SINX + COSY COSX
8

WHERE X = OUTER GIMBAL ANGLE
Y = INNER GIMBAL ANGLE
Z = MIDDLE GIMBAL ANGLE

THE INTERPRETATION OF THIS MATRIX IS AS FOLLOWS:

IF A_X, A_Y, A_Z REPRESENT THE COMPONENTS OF A VECTOR IN S/C AXES THEN THE COMPONENTS OF THE SAME VECTOR IN

STABLE MEMBER AXES (B_X, B_Y, B_Z) ARE

[B_X] [A_X]
[X] [X]
[] []
[B_Y] * [A_Y]
[Y] = M [Y]
[] []
[B_Z] [B_Z]
[Z] [Z]
#

THE SUBROUTINE WILL STORE THIS MATRIX IN SEQUENTIAL LOCATIONS OF ERASABLE MEMORY AS SPECIFIED BY THE CALLING
PROGRAM. TO DO THIS THE CALLING PROGRAM MUST FIRST LOAD X2 WITH THE COMPLEMENT OF THE STARTING ADDRESS FOR M.
#

INTERNALLY, THE ROUTINE USES THE FIRST 16 LOCATIONS OF THE PUSH DOWN LIST, ALSO STEP REGISTER S1 AND INDEX
REGISTER X2.

DCM TO CDU

THIS ROUTINE EXTRACTS THE CDU ANGLES FROM A DIRECTION COSINE MATRIX (M SCALED BY 2) RELATING S/C AXIS TO
STABLE MEMBER AXES. X1 MUST CONTAIN THE COMPLEMENT OF THE STARTING ADDRESS FOR M. THE SUBROUTINE LEAVES THE
CORRESPONDING GIMBAL ANGLES IN V(MPAC) AS DOUBLE PRECISION 1'S COMPLEMENT ANGLES SCALED BY 2PI. THE FORMULAS
FOR THIS CONVERSION ARE

Z = ARCSIN (M / 3)
#

Y = ARCSIN (-M / COSZ)
6

IF M₀ IS NEGATIVE, Y IS REPLACED BY PI SGN Y - Y.
#

ATTITUDE_MANEUVER_ROUTINE

```
#          X          =      ARCSIN ( -M /COSZ)
#                                     5
#
# IF M  IS NEGATIVE, X IS REPLACED BY PI SGN X - X.
# 4
#
# THIS ROUTINE DOES NOT SET THE PUSH DOWN POINTER, BUT USES THE NEXT 8 LOCATIONS OF THE PUSH DOWN LIST AND
# RETURNS THE POINTER TO ITS ORIGINAL SETTING.  THIS PROCEDURE ALLOWS THE CALLER TO STORE THE MATRIX AT THE TOP OF
# THE PUSH DOWN LIST.
#
# DELCOMP
# -----
#
# *
# THIS ROUTINE COMPUTES THE DIRECTION COSINE MATRIX (DEL) RELATING ON
#
# IS ROTATED WITH RESPECT TO THE FIRST BY AN ANGLE, A, ABOUT A UNIT VECTOR Ū.  THE FORMULA FOR THIS MATRIX IS
#
# *
# DEL      =      I COSA + Ū ŪT (1 - COSA) + VX* SINA
#
# WHERE
#
# I      =      [ 1  0  0 ]
#                [ 0  1  0 ]
#                [ 0  0  1 ]
#
#
#                [ 2
#                [  U      U  U      U
#                [  X      X  Y      X  Z ]
#                [
#
# Ū ŪT =      [ 2
#                [  U  U      U      U
#                [  Y  X      Y      Y  Z ]
#                [
#                [  U  U      U      U      2
#                [  Z  X      Z  Y      Z ]
#
#
#                [  0      -U      U ]
#                [      Z      Y ]
#                [
#
# VX* =      [  U      0      -U ]
#                [  Z      X ]
#                [
#                [ -U      U      0 ]
#                [  Y      X ]
```

1412THE
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

```
#
#      U      =      UNIT ROTATION VECTOR RESOLVED INTO S/C AXES.
#      A      =      ROTATION ANGLE
#
#      *
# THE INTERPRETATION OF DEL IS AS FOLLOWS:
#
# IF AX , AY , AZ REPRESENT THE COMPONENTS OF A VECTOR IN THE ROTATED FRAME, THEN THE COMPONENTS OF THE SAME
# VECTOR IN THE ORIGINAL S/C AXES (BX , BY , BZ) ARE
#
#      [ BX ]      [ AX ]
#      [ X  ]      [ X  ]
#      [ BY ]      [ AY ]
#      [ Y  ]      [ Y  ]
#      [ BZ ]      [ BZ ]
#      [ Z  ]      [ Z  ]
#
#      *
#      DEL
#
# THE ROUTINE WILL STORE THIS MATRIX (SCALED UNITY) IN SEQUENTIAL LOCATIONS OF ERASABLE MEMORY BEGINNING WITH
# THE LOCATION CALLED DEL.  IN ORDER TO USE THE ROUTINE, THE CALLING PROGRAM MUST FIRST STORE U (A HALF UNIT
# DOUBLE PRECISION VECTOR) IN THE SET OF ERASABLE LOCATIONS BEGINNING WITH THE ADDRESS CALLED COF.  THE ANGLE, A,
# MUST THEN BE LOADED INTO D(MPAC).
#
# INTERNALLY, THE PROGRAM ALSO USES THE FIRST 10 LOCATIONS OF THE PUSH DOWN LIST.
#
# READCDUK
# -----
#
# THIS BASIC LANGUAGE SUBROUTINE LOADS T(MPAC) WITH THE THREE CDU ANGLES.
#
# SIGNMPAC
# -----
#
# THIS IS A BASIC LANGUAGE SUBROUTINE WHICH LIMITS THE MAGNITUDE OF D(MPAC) TO + OR - DPOSMAX ON OVERFLOW.
#
# PROGRAM STORAGE ALLOCATION
#
#      1)      FIXED MEMORY      1059 WORDS
#      2)      ERASABLE MEMORY    98
#      3)      STATE SWITCHES     3
```

ATTITUDE_MANEUVER_ROUTINE

4) FLAGS 1

JOB PRIORITIES

1) KALCMANU TBD

2) ONE SECOND UPDATE TBD

SUMMARY OF STATE SWITCHES AND FLAGWORDS USED BY KALCMANU.

STATE SWITCH NO.	FLAGWRD 2 BIT NO.	SETTING	MEANING
*			
31	14	0	MANEUVER WENT THROUGH GIMBAL LOCK
		1	MANEUVER DID NOT GO THROUGH GIMBAL LOCK
*			
32	13	0	CONTINUE UPDATE PROCESS
		1	START UPDATE PROCESS
33	12	0	PERFORM FINAL P AXIS YAW IF REQUIRED
		1	IGNORE ANY FINAL P-AXIS YAW
34	11	0	SIGNAL END OF KALCMANU
		1	KALCMANU IN PROCESS. USER MUST SET SWITCH BEFORE INITIATING

* INTERNAL TO KALCMANU

SUGGESTIONS FOR PROGRAM INTEGRATION

THE FOLLOWING VARIABLES SHOULD BE ASSIGNED TO UNSWITCH ERASABLE:

CPHI
CTHETA
CPSI
POINTVSM +5
SCAXIS +5
DELDCDU
DELDCDU1
DELDCDU2
RATEINDX

THE FOLLOWING SUBROUTINES MAY BE PUT IN A DIFFERENT BANK

MXM3



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

TRANSPGS
SIGNMPAC
READCDUK
CDUTODCM

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60


```
1
2      BANK      15
3      SETLOC    KALCMON1
4      BANK
5
6      EBANK=     BCDU
7
8      # THE THREE DESIRED CDU ANGLES MUST BE STORED AS SINGLE PRECISION TWO'S COMPLEMENT ANGLES IN THE THREE SUCCESSIVE
9      # LOCATIONS, CPHI, CTHETA, CPSI.
10
11      KALCMAN3   COUNT*  $$/KALC
12                TC      INTPRET      # PICK UP THE CURRENT CDU ANGLES AND
13                RTB      #           COMPUTE THE MATRIX FROM INITIAL S/C
14                READCDUK #           AXES TO FINAL S/C AXES.
15                STORE    BCDU        # STORE INITIAL S/C ANGLES
16                SLOAD    ABS          # CHECK THE MAGNITUDE OF THE DESIRED
17                CPSI     # MIDDLE GIMBAL ANGLE
18                DSU      BPL
19                LOCKANGL # IF GREATER THAN 70 DEG ABORT MANEUVER
20                TOOBADF
21                AXC,2    TLOAD
22                MIS
23                BCDU
24                CALL     # COMPUTE THE TRANSFORMATION FROM INITIAL
25                AXC,2    CDUTODCM     # S/C AXES TO STABLE MEMBER AXES
26                TLOAD
27                MFS      # PREPARE TO CALCULATE ARRAY MFS
28                CPHI
29                CALL
30                AXC,1    CDUTODCM
31      SECAD      AXC,1    CALL          # MIS AND MFS ARRAYS CALCULATED      $2
32                MIS
33                TRANSPOS
34                VLOAD    STADR
35                STOVL     TMIS +12D
36                STADR
37                STOVL     TMIS +6
38                STADR
39                STORE     TMIS          # TMIS = TRANSPOSE(MIS) SCALED BY 2
40                AXC,1    AXC,2
41                TMIS
42                MFS
43                CALL
44                VLOAD    MXM3
45                STADR
46                STOVL     MFI +12D
47                STADR
48                STOVL     MFI +6
49                STADR
50                STORE     MFI          # MFI = TMIS MFS (SCALED BY 4)
51                SETPD     CALL        # TRANSPOSE MFI IN PD LIST
52
53
54
55
56
57
58
59
60
```

18D
TRNSPSPDVLOAD STADR
STOVL TMFI +12D
STADRSTOVL TMFI +6
STADR
STORE TMFI

TMFI = TRANSPOSE (MFI) SCALED BY 4

CALCULATE COFSKEW AND MFISYM

DLOAD DSU
TMFI +2
MFI +2PDDL DSU # CALCULATE COF SCALED BY 2/SIN(AM)
MFI +4
TMFI +4PDDL DSU
TMFI +10D
MFI +10DVDEF
STORE COFSKEW # EQUALS MFISKEW

CALCULATE AM AND PROCEED ACCORDING TO ITS MAGNITUDE

DLOAD DAD
MFI
MFI +16D
DSU DADDP1/4TH
MFI +8D

STORE CAM # CAM = (MFI0+MFI4+MFI8-1)/2 HALF SCALE

ARCCOS
STORE AM # AM=ARCCOS(CAM) (AM SCALED BY 2)
DSU BPLMINANG
CHECKMAX

TLOAD # MANEUVER LESS THAN .25 DEGREES

STCALL CPHI # GO DIRECTLY INTO ATTITUDE HOLD
CDUXD # ABOUT COMMANDED ANGLES
TOOBADI # STOP RATE AND EXIT

CHECKMAX

DLOAD DSU
AM
BPL MAXANG
VLOAD
ALTCALC # UNIT
COFSKEW # COFSKEWUNIT
STORE COF # COF IS THE MANEUVER AXIS

1					1
2		GOTO		# SEE IF MANEUVER GOES THRU GIMBAL LOCK	2
3			LOCSKIRT		3
4	ALTCALC	VLOAD	VAD	# IF AM GREATER THAN 170 DEGREES	4
5			MFI		5
6			TMFI		6
7		VSR1			7
8		STOVL	MFISYM		8
9			MFI	+6	9
10		VAD	VSR1		10
11			TMFI	+6	11
12		STOVL	MFISYM	+6	12
13			MFI	+12D	13
14		VAD	VSR1		14
15			TMFI	+12D	15
16		STORE	MFISYM	+12D # MFISYM=(MFI+TMFI)/2 SCALED BY 4	16
17					17
18	# CALCULATE COF				18
19					19
20		DLOAD	SR1		20
21			CAM		21
22		PDDL	DSU	# PDO CAM \$4	22
23			DPHALF		23
24			CAM		24
25		BOVB	PDDL	# PS2 1 - CAM \$2	25
26			SIGNMPAC		26
27			MFISYM	+16D	27
28		DSU	DDV		28
29			0		29
30			2		30
31		SQRT	PDDL	# COFZ = SQRT(MFISYM8-CAM)/(1-CAM)	31
32			MFISYM	+8D # \$ ROOT 2	32
33		DSU	DDV		33
34			0		34
35			2		35
36		SQRT	PDDL	# COFY = SQRT(MFISYM4-CAM)/(1-CAM) \$ROOT2	36
37			MFISYM		37
38		DSU	DDV		38
39			0		39
40			2		40
41		SQRT	VDEF	# COFX = SQRT(MFISYM-CAM)/(1-CAM) \$ROOT 2	41
42		UNIT			42
43		STORE	COF		43
44					44
45	# DETERMINE LARGEST COF AND ADJUST ACCORDINGLY				45
46					46
47	COFMAXGO	DLOAD	DSU		47
48			COF		48
49			COF	+2	49
50		BMN	DLOAD	# COFY G COFX	50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1					
2			COMP12		
3			COF		
4		DSU	BMN		
5			COF	+4	
6			METHOD3		# COFZ G COFX OR COFY
7		GOTO			
8			METHOD1		# COFX G COFY OR COFZ
9	COMP12	DLOAD	DSU		
10			COF	+2	
11			COF	+4	
12		BMN			
13			METHOD3		# COFZ G COFY OR COFX
14					
15	METHOD2	DLOAD	BPL		# COFY MAX
16			COFSKEW	+2	# UY
17			U2POS		
18		VLOAD	VCOMP		
19			COF		
20		STORE	COF		
21	U2POS	DLOAD	BPL		
22			MFISYM	+2	# UX UY
23			OKU21		
24		DLOAD	DCOMP		# SIGN OF UX OPPOSITE GARBLED
25			COF		
26		STORE	COF		
27	OKU21	DLOAD	BPL		
28			MFISYM	+10D	# UY UZ
29			LOCSKIRT		
30		DLOAD	DCOMP		# SIGN OF UZ OPPOSITE TO UY
31			COF	+4	
32		STORE	COF	+4	
33		GOTO			
34			LOCSKIRT		
35	METHOD1	DLOAD	BPL		# COFX MAX
36			COFSKEW		# UX
37			UIPOS		
38		VLOAD	VCOMP		
39			COF		
40		STORE	COF		
41	UIPOS	DLOAD	BPL		
42			MFISYM	+2	# UX UY
43			OKU12		
44		DLOAD	DCOMP		
45			COF	+2	# SIGN OF UY OPPOSITE TO UX
46		STORE	COF	+2	
47	OKU12	DLOAD	BPL		
48			MFISYM	+4	# UX UZ
49			LOCSKIRT		
50		DLOAD	DCOMP		# SIGN OF UZ OPPOSITE TO UY
51			COF	+4	
52					
53					
54					
55					
56					
57					
58					
59					
60					

MATRIX OPERATIONS

BANK 13
SETLOC KALCMON2
BANK

EBANK= BCDU

MXM3 SETPD VLOAD* # MXM3 MULTIPLIES 2 3X3 MATRICES
0 # AND LEAVES RESULT IN PD LIST
0,1 # AND MPAC

VXM* PDVL*
0,2
6,1

VXM* PDVL*
0,2
12D,1

VXM* PUSH
0,2

RVQ

RETURN WITH MIXM2 IN PD LIST

TRANSPOS SETPD VLOAD* # TRANSPOS TRANSPOSES A 3X3 MATRIX
0 # AND LEAVES RESULT IN PD LIST
0,1 # MATRIX ADDRESS IN XR1

PDVL* PDVL*
6,1
12D,1

TRNSPSPD PUSH # MATRIX IN PD
EXIT # ENTER WITH MATRIX AT 0 IN PD LIST
INDEX

FIXLOC

DXCH 12
INDEX FIXLOC
DXCH 16

INDEX FIXLOC
DXCH 12
INDEX FIXLOC

DXCH 14
INDEX FIXLOC
DXCH 4

INDEX FIXLOC
DXCH 14
INDEX FIXLOC

DXCH 2
INDEX FIXLOC
DXCH 6

INDEX FIXLOC
DXCH 2

```
1
2          TC      INTPRET
3          RVQ
4
5          BANK    15
6          SETLOC  KALCMON1
7          BANK
8
9          EBANK=   BCDU
10
11         MINANG   2DEC    0.00069375
12
13         MAXANG   2DEC    0.47222222
14
15         # GIMBAL LOCK CONSTANTS
16
17         # D = MGA CORRESPONDING TO GIMBAL LOCK = 60 DEGREES
18         #       NGL = BUFFER ANGLE (TO AVOID DIVISIONS BY ZERO) = 2 DEGREES
19
20         SD        2DEC    .433015      # = SIN(D)          $2
21
22         K3S1      2DEC    .86603       # = SIN(D)          $1
23
24         K4        2DEC    -.25         # = -COS(D)        $2
25
26         K4SQ      2DEC    .125         # = COS(D)COS(D)   $2
27
28         SNGLCD    2DEC    .008725      # = SIN(NGL)COS(D) $2
29
30         CNGL      2DEC    .499695      # COS(NGL)         $2
31
32         LOCKANGL  DEC     .388889      # = 70 DEGREES
33
34         # INTERPRETIVE SUBROUTINE TO READ THE CDU ANGLES
35
36         READCDUK  CA      CDUZ          # LOAD T(MPAC) WITH CDU ANGLES
37
38         TS        MPAC    +2
39         EXTEND
40         DCA       CDUX          # AND CHANGE MODE TO TRIPLE PRECISION
41         TCF       TLOAD    +6
42
43         CDUTODCM  AXT,1  SSP
44         OCT       3
45         S1
46         OCT       1          # SET XR1, S1, AND PD FOR LOOP
47         STORE    7
48         SETPD
49         0
50         LOOPSIN   SLOAD*  RTB
51                   10D,1
52                   CDULOGIC
```

```
1
2      STORE 10D      # LOAD PD WITH 0 SIN(PHI)
3      SIN   PDDL      #                2 COS(PHI)
4
5      COS   10D      #                4 SIN(THETA)
6      TIX,1 PUSH      #                6 COS(THETA)
7      DLOAD      #                8 SIN(PSI)
8      LOOPSIN      #                10 COS(PSI)
9
10     DMP 6
11     SL1
12
13     STORE 10D
14     DLOAD 0,2      # C0 = COS(THETA)COS(PSI)
15     DMP
16
17     PDDL 4
18     DMP 0          # (PD6 SIN(THETA)SIN(PHI))
19
20     DMP 6
21     8D
22     SL1
23
24     BDSU 2
25     SL1
26     12D
27
28     STORE 2,2      # C1=-COS(THETA)SIN(PSI)COS(PHI)
29     DLOAD DMP
30     2
31
32     PDDL 4
33     DMP 6          # (PD7 COS(PHI)SIN(THETA)) SCALED 4
34
35     DMP 8D
36     SL1
37     0
38
39     DAD SL1
40     14D
41
42     STORE 4,2      # C2=COS(THETA)SIN(PSI)SIN(PHI)
43     DLOAD
44
45     STORE 8D
46     6,2          # C3=SIN(PSI)
47     DLOAD
48
49     DMP 10D
50     SL1
51
52     STORE 2
53     8D,2          # C4=COS(PSI)COS(PHI)
54     DLOAD DMP
55
56     DCOMP 10D
57     SL1
58
59     STORE 10D,2    # C5=-COS(PSI)SIN(PHI)
60     DLOAD DMP
61     4
62
63     DCOMP 10D
64     SL1
65
66     STORE 12D,2    # C6=-SIN(THETA)COS(PSI)
67
```



```
1      DLOAD
2      DMP      SL1      # (PUSH UP 7)
3
4      8D
5      PDDL      DMP      # (PD7 COS(PHI)SIN(THETA)SIN(PSI)) SCALE 4
6      6
7
8      0
9      DAD      SL1      # (PUSH UP 7)
10     STADR      # C7=COS(PHI)SIN(THETA)SIN(PSI)
11     STORE      14D,2    #      +COS(THETA)SIN(PHI)
12     DLOAD
13     DMP      SL1      # (PUSH UP 6)
14     8D
15     PDDL      DMP      # (PD6 SIN(THETA)SIN(PHI)SIN(PSI)) SCALE 4
16     6
17
18     2
19     DSU      SL1      # (PUSH UP 6)
20     STADR
21     STORE      16D,2    # C8=-SIN(THETA)SIN(PHI)SIN(PSI)
22     RVQ      # +COS(THETA)COS(PHI)
```

CALCULATION OF THE MATRIX DEL.....

```
23 #
24 #      *      *      T      *
25 #      DEL = (IDMATRIX)COS(A)+UU (1-COS(A))+UX SIN(A)      SCALED 1
26 #
27 #      WHERE  $\bar{U}$  IS A UNIT VECTOR (DP SCALED 2) ALONG THE AXIS OF ROTATION.
28 #      A IS THE ANGLE OF ROTATION (DP SCALED 2)
29 #
30 #      UPON ENTRY, THE STARTING ADDRESS OF  $\bar{U}$  IS COF, AND A IS IN MPAC
```

```
31
32 DELCOMP      SETPD      PUSH      # MPAC CONTAINS THE ANGLE A
33      0
34      SIN      PDDL      # PD0 = SIN(A)
35      COS      PUSH      # PD2 = COS(A)
36      SR2      PDDL      # PD2 = COS(A)      $8
37      BDSU      BOVB
38      DPHALF
39      SIGNMPAC
40      PDDL      # PDA = 1-COS(A)
```

COMPUTE THE DIAGONAL COMPONENTS OF DEL

```
43
44      DSQ      COF
45      DMP
46      4
47      DAD      SL3
48      2
49      BOVB
50      SIGNMPAC
```

1						1
2		STODL	KEL		# UX UX(1-COS(A)) +COS(A)	2
3			COF	+2		3
4		DSQ	DMP			4
5			4			5
6		DAD	SL3			6
7			2			7
8		BOVB				8
9			SIGNMPAC			9
10		STODL	KEL	+8D	# UY UY(1-COS(A)) +COS(A)	10
11			COF	+4		11
12		DSQ	DMP			12
13			4			13
14		DAD	SL3			14
15			2			15
16		BOVB				16
17			SIGNMPAC			17
18		STORE	KEL	+16D	# UZ UZ(1-COS(A)) +COS(A)	18
19						19
20	# COMPUTE THE OFF DIAGONAL TERMS OF DEL					20
21						21
22		DLOAD	DMP			22
23			COF			23
24			COF	+2		24
25		DMP	SL1			25
26			4			26
27		PDDL	DMP		# D6 UX UY (1-COS A)	27
28			COF	+4		28
29			0			29
30		PUSH	DAD		# D8 UZ SIN A	30
31			6			31
32		SL2	BOVB			32
33			SIGNMPAC			33
34		STODL	KEL	+6		34
35		BDSU	SL2			35
36		BOVB				36
37			SIGNMPAC			37
38		STODL	KEL	+2		38
39			COF			39
40		DMP	DMP			40
41			COF	+4		41
42			4			42
43		SL1	PDDL		# D6 UX UZ (1-COS A)	43
44			COF	+2		44
45		DMP	PUSH		# D8 UY SIN(A)	45
46			0			46
47		DAD	SL2			47
48			6			48
49		BOVB				49
50			SIGNMPAC			50
51		STODL	KEL	+4	# UX UZ (1-COS(A))+UY SIN(A)	51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

BDSU
BOVB

SL2

STODL

SIGNMPAC

KEL

+12D

UX UZ (1-COS(A))-UY SIN(A)

COF

+2

DMP

DMP

COF

+4

SL1

PDDL

D6

UY UZ (1-COS(A))

\$ 4

COF

DMP

PUSH

D8

UX SIN(A)

DAD

SL2

6

BOVB

SIGNMPAC

STODL

KEL

+14D

UY UZ(1-COS(A)) +UX SIN(A)

BDSU

SL2

BOVB

SIGNMPAC

STORE

KEL

+10D

UY UZ (1-COS(A)) -UX SIN(A)

RVQ

DIRECTION COSINE MATRIX TO CDU ANGLE ROUTINE

X1 CONTAINS THE COMPLEMENT OF THE STARTING ADDRESS FOR MATRIX (SCALED 2).

LEAVE CDU ANGLES SCALED 2PI IN V(MPAC).

COS(MGA) WILL BE LEFT IN S1 (SCALED 1).

#

THE DIRECTION COSINE MATRIX RELATING S/C AXES TO STABLE MEMBER AXES CAN BE WRITTEN AS:

#

C = COS(THETA) COS(PSI

0

#

C = -COS(THETA) SIN(PSI) COS(PHI) + SIN(THETA) SIN(PHI)

1

#

C = COS(THETA) SIN(PSI) SIN(PHI) + SIN(THETA) COS(PHI)

2

#

C = SIN(PSI)

3

#

C = COS(PSI) COS(PHI)

4

#

C = -COS(PSI) SIN(PHI)

5

#

C = -SIN(THETA) COS(PSI)

6

#

C = SIN(THETA) SIN(PSI) COS(PHI) + COS (THETA) SIN(PHI)

7

#

C = -SIN(THETA) SIN(PSI) SIN(PHI) + COS(THETA)COS(PHI)

8

#

```
#
# WHERE PHI = OGA
#      THETA = IGA
#      PSI = MGA

DCMTOCDU      DLOAD*  ARCSIN
                6,1
                PUSH   COS          # PD +0      PSI
                SL1    BOVB
                STORE   S1
                DLOAD*  DCOMP
                DDV     12D,1
                DDV     ARCSIN
                S1
                PDDL*   BPL          # PD +2      THETA
                0,1      # MUST CHECK THE SIGN OF COS(THETA)
                        # TO DETERMINE THE PROPER QUADRANT.
                DLOAD   OKTHETA
                BPL     DCOMP
                        DAD
                        SUHALFA
                        DPHALF
                GOTO
SUHALFA      DSU      CALCPHI
                DPHALF
CALCPHI      PUSH
OKTHETA      DLOAD*   DCOMP
                10D,1
                DDV     ARCSIN
                S1
                PDDL*   BPL          # PUSH DOWN PHI
                8D,1
                DLOAD   OKPHI
                BPL     DCOMP          # PUSH UP PHI
                DAD
                SUHALFAP
                DPHALF
                GOTO
SUHALFAP      DSU      VECOFANG
                GOTO
                DPHALF
                VECOFANG
OKPHI      DLOAD          # PUSH UP PHI
VECOFANG    VDEF     RVQ
```

ROUTINES FOR TERMINATING THE AUTOMATIC MANEUVER AND RETURNING TO USER.

TOOBADF EXIT ALARM
 TC 00401
 OCT
 TCF NOGO # DO NOT ZERO ATTITUDE ERRORS

 TC BANKCALL
 CADR ZATTEROR # ZERO ATTITUDE ERRORS

NOGO TC BANKCALL
 CADR STOPRATE # STOP RATES

 CAF TWO
 INHINT # ALL RETURNS ARE NOW MADE VIA GOODEND
 TC WAITLIST

 EBANK= BCDU
 2CADR GOODMANU

 TCF ENDOFJOB

TOOBADI EXIT
 TCF NOGO

BANK 15

SETLOC KALCMON1
BANK# DETECTING GIMBAL LOCK
LOCSKIRT EQUALS NOGIMLOC

NOGIMLOC SET

WCALC LXC,1 CALCMAN3
DLOAD*SR4 RATEINDX # CHOOSE THE DESIRED MANEUVER RATE
ARATE,1 # FROM A LIST OF FOUR
CALL # COMPUTE THE INCREMENTAL ROTATION MATRIX
DELCOMP # DEL CORRESPONDING TO A 1 SEC ROTATION
ABOUT COF

DLOAD* VXSC

ARATE,1
COF

STODL BRATE # COMPONENT MANEUVER RATES 45 DEG/SEC

DMP AM
DDV*

ANGLTIME

ARATE,1

SR

5

STORE TM

SETGO

CALCMAN2

NEWANGL +1

MANEUVER EXECUTION TIME SCALED AS T2

0(OFF) = CONTINUE MANEUVER

1(ON) = START MANEUVER

THE FOUR SELECTABLE FREE FALL MANEUVER RATES SELECTED BY

LOADING RATEINDX WITH 0,2,4,6, RESPECTIVELY

ARATE 2DEC .0088888888 # = 0.2 DEG/SEC \$ 22.5 DEG/SEC

2DEC .0222222222 # = 0.5 DEG/SEC \$ 22.5 DEG/SEC

2DEC .0888888888 # = 2.0 DEG/SEC \$ 22.5 DEG/SEC

2DEC .4444444444 # = 10.0 DEG/SEC \$ 22.5 DEG/SEC

ANGLTIME 2DEC .0001907349 # = 1008-19 FUDGE FACTOR TO CONVERT
MANEUVER ANGLE TO MANEUVER TIME

```
1 # GENERATION OF STEERING COMMANDS FOR DIGITAL AUTOPILOT FREE FALL MANEUVERS
2 #
3 # NEW COMMANDS WILL BE GENERATED EVERY ONE SECOND DURING THE MANEUVER
4
5         EBANK=  TTEMP
6
7 NEWDELHI      TC      BANKCALL      # CHECK FOR AUTO STABILIZATION
8 CADR          ISITAUTO  # ONLY
9
10        CCS      A
11        TCF      NOGO -2
12 NEWANGL      TC      INTPRET
13 AXC,1        AXC,2
14 MIS          # COMPUTE THE NEW MATRIX FROM S/C TO
15 KEL          # STABLE MEMBER AXES
16
17        CALL
18        VLOAD    MXM3
19        STOVL    STADR
20        STOVL    MIS +12D      # CALCULATE NEW DESIRED CDU ANGLES
21        STOVL    MIS +6D
22        STADR
23        STORE    MIS
24        AXC,1    CALL
25        MIS
26        DCMTOCDU      # PICK UP THE NEW CDU ANGLES FROM MATRIX
27        RTB
28        VISTO2S
29        STORE    NCDU      # NEW CDU ANGLES
30        BONCLR   EXIT
31        CALCMAN2
32        MANUSTAT      # TO START MANEUVER
33        CAF      TWO      # +0 OTHERWISE
34 INCRDCDU     TS      SPNDX
35 INDEX        SPNDX
36 CA           BCDU      # INITIAL CDU ANGLES
37 EXTEND       # OR PREVIOUS DESIRED CDU ANGLES
38 INDEX        SPNDX
39 MSU          NCDU
40
41        EXTEND
42        SETLOC   KALCMON1
43        BANK
44        MP      DT/TAU
45        CCS      A      # CONVERT TO 2S COMPLEMENT
46        AD      ONE
47        TCF      +2
48        COM
49        INDEX    SPNDX
50        TS      DELDCDU      # ANGLE INCREMENTS TO BE ADDED TO
51        INDEX    SPNDX      # CDUXD, CDUYD, CDUZD EVERY TENTH SECOND
```

```
1
2      CA      NCDU      # BY LEM DAP
3      INDEX   SPNDX
4      XCH     BCDU
5      INDEX   SPNDX
6      TS      CDUXD
7      CCS     SPNDX
8      TCF     INCRDCDU      # LOOP FOR THREE AXES
9
10     RELINT
11
12     # COMPARE PRESENT TIME WITH TIME TO TERMINATE MANEUVER
13
14     TMANUCHK      TC      TIMECHK
15                  TCF     CONTMANU
16
17     MANUSTAL      CAF     ONE
18                  INHINT   # END MAJOR PART OF MANEUVER WITHIN 1 SEC
19                  TC      WAITLIST      # UNDER WAITLIST CALL TO MANUSTOP
20
21     EBANK=      TTEMP
22     2CADR      MANUSTOP
23
24     RELINT
25     TCF     ENDOFJOB
26
27     TIMECHK      EXTEND
28                  DCS     TIME2
29                  DXCH    TTEMP
30
31     EXTEND
32     DCA      TM
33     DAS      TTEMP
34
35     CCS      TTEMP
36     TC      Q
37     TCF      +2
38
39     TCF      2NDRETRN
40     CCS      TTEMP +1
41     TC      Q
42
43     TCF      MANUOFF
44
45     MANUOFF      COM
46                  AD      ONESEK +1
47
48     EXTEND
49     BZMF      2NDRETRN
50     INCR      Q
51
52     2NDRETRN      INCR      Q
53                  TC      Q
54
55     DT/TAU      DEC      .1
56
57     MANUSTAT      EXIT      # INITIALIZATION ROUTINE
58                  EXTEND    # FOR AUTOMATIC MANEUVERS
59                  DCA      TIME2
60
```


	DAS	TM	# TM+TO	MANEUVER COMPLETION TIME
	EXTEND			
	DCS	ONESEK		
	DAS	TM	# (TM+TO)-1	
	INHINT			
RATEBIAS	CAF	TWO		
	TS	KSPNDX		
	DOUBLE			
	TS	KDPNDX		
	INDEX	A		
	CA	BRATE		
	INDEX	KSPNDX	# STORE MANEUVER RATE IN	
	TS	OMEGAPD	# OMEGAPD, OMEGAQD, OMEGARD	
	EXTEND			
	BZMF	+2	# COMPUTE ATTITUDE ERROR	
	COM		# OFFSET = (WX)ABS(WX)/2AJX	
	EXTEND		# WHERE AJX= 2-JET ACCELERATION	
	MP	BIASCALE	# = -1/16	
	EXTEND			
	INDEX	KDPNDX		
	MP	BRATE		
	EXTEND			
	INDEX	KSPNDX		
	DV	1JACC	# =AJX \$ 90 DEG/SEC-SEC	
	INDEX	KSPNDX		
	TS	DELPORR	# \$ 180 DEG	
	CCS	KSPNDX		
	TCF	RATEBIAS		
	CA	TIME1		
	AD	ONESEK +1		
	XCH	NEXTIME		
	TCF	INCRDCDU -1		
ONESEK	DEC	0		
	DEC	100		
BIASCALE	OCT	75777	# = -1/16	
CONTMANU	CS	TIME1	# RESET FOR NEXT DCDU UPDATE	
	AD	NEXTIME		
	CCS	A		
	AD	ONE		
	TCF	MANUCALL		
	AD	NEGMAX		
MANUCALL	COM			
	INHINT		# CALL FOR NEXT UPDATE VIA WAITLIST	
	TC	WAITLIST		
	EBANK=	TTEMP		
	2CADR	UPDTCALL		



1					1
2		CAF	ONESEK +1	# INCREMENT TIME FOR NEXT UPDATE	2
3		ADS	NEXTIME		3
4		TCF	ENDOFJOB		4
5					5
6	UPDTCALL	CAF	PRI026	# SATELLITE PROGRAM TO CALL FOR UPDATE	6
7		TC	FINDVAC	# OF STEERING COMMANDS	7
8		EBANK=	TTEMP		8
9		2CADR	NEWDELHI		9
10					10
11		TC	TASKOVER		11
12					12
13					13
14					14
15					15
16					16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

KALCMANU_STEERING

ROUTINE FOR TERMINATING AUTOMATIC MANEUVERS

MANUSTOP	CAF	ZERO	# ZERO MANEUVER RATES
	TS	DELDCDU2	
	TS	OMEGARD	
	TS	DELRREROR	
	TS	DELDCDU1	
	TS	OMEGAQD	
	TS	DELQEROR	
	CA	CPSI	# SET DESIRED GIMBAL ANGLES TO
	TS	CDUZD	# DESIRED FINAL GIMBAL ANGLES
	CA	CTHETA	
ENDROLL	TS	CDUYD	
	CA	CPHI	# NO FINAL YAW
	TS	CDUXD	
	CAF	ZERO	
	TS	OMEGAPD	# I.E. MANEUVER DID NOT GO THRU
	TS	DELDCDU	# GIMBAL LOCK ORIGINALLY
GOODMANU	TS	DELPEROR	
	CA	ATTPRIO	# RESTORE USERS PRIO
	TS	NEWPRIO	
	CA	ZERO	# ZERO ATTCADR
	DXCH	ATTCADR	
	TC	SPVAC	# RETURN TO USER
	TC	TASKOVER	

EBANK= XSM

BANK 33
SETLOC E/PROG
BANK

COUNT* \$\$/P07

SPECIAL PROGRAMS TO EASE THE PANGS OF ERASABLE MEMORY PROGRAMS.

#

E/BKCALL FOR DOING BANKCALLS FROM AND RETURNING TO ERASABLE.

#

THIS ROUTINE IS CALLABLE FROM ERASABLE OR FIXED. LIKE BANKCALL, HOWEVER, SWITCHING BETWEEN S3 AND S4
IS NOT POSSIBLE.

#

THE CALLING SEQUENCE IS:

#

TC BANKCALL
CADR E/BKCALL
CADR ROUTINE # WHERE TO WANT TO GO IN FIXED.

RETURN HERE FROM DISPLAY TERMINATE, BAD STALL OR TC Q.

RETURN HERE FROM DISPLAY PROCEED OR GOOD RETURN FROM STALL.

RETURN HERE FROM DISPLAY ENTER OR RECYCLE.

#

THIS ROUTINE REQUIRES TWO ERASABLES (EBUF2, +1) IN UNSWITCHED WHICH ARE UNSHARED BY INTERRUPTS AND
OTHER EMEMORY PROGRAMS.

#

A + L ARE PRESERVED THROUGH BANKCALL AND E/BKCALL.

E/BKCALL DXCH BUF2 # SAVE A,L AND GET DP RETURN.
DXCH EBUF2 # SAVE DP RETURN.
INCR EBUF2 # RETURN +1 BECAUSE DOUBLE CADR.CA BBANK
MASK LOW10 # GET CURRENT EBANK. (SBANK SOMEDAY)
ADS EBUF2 +1 # FORM BBCON. (WAS FBANK)NDX EBUF2
CA 0 -1 # GET CADR OF ROUTINE.
TC SWCALL # GO TO ROUTINE, SETTING Q TO SWRETURN
AND RESTORING A + L.TC +4 # TX Q, V34, OR BAD STALL RETURN.
TC +2 # PROCEED OR GOOD STALL RETURN.

INCR EBUF2 # ENTER OR RECYCLE RETURN.

E/SWITCH INCR EBUF2
DXCH EBUF2

DTCB

SYSTEM_TEST_STANDARD_LEAD_INS

```
# E/CALL      FOR CALLING A FIXED MEMORY INTERPRETIVE SUBROUTINE FROM ERASABLE AND RETURNING TO ERASABLE.
#
# THE CALLING SEQUENCE IS...
#
#      RTB
#
#      CADR      E/CALL
#                  ROUTINE      # THE INTERPRETIVE SUBROUTINE YOU WANT.
#                                # RETURNS HERE IN INTERPRETIVE.
#
# E/CALL      LXCH      LOC      # ADRES -1 OF CADR.
#              INDEX    L
#
#              CA        L      # CADR IN A.
#              INCR      L
#              INCR      L      # RETURN ADRES IN L.
#              DXCH      EBUF2   # STORE CADR AND RETURN
#              TC        INTPRET
#              CALL
#
#              EXIT      EBUF2   # INDIRECTLY EXECUTE ROUTINE. IT MUST
#              LXCH      EBUF2   # LEAVE VIA RVQ OR EQUIVALENT.
#              TCF       INTPRET +1 # PICK UP RETURN.
#                                # SET LOC AND RETURN TO CALLER.
```

SYSTEM_TEST_STANDARD_LEAD_INS

```
# E/JOBWAK      FOR WAKING UP ERASABLE MEMORY JOBS.
#
# THIS ROUTINE MUST BE CALLED IN INTERRUPT OR WITH INTERRUPTS INHIBITED.
# THE CALLING SEQUENCE IS:
#
#      INHINT
#      ...
#      CA      WAKEADR      # ADDRESS OF SLEEPING JOB
#      TC      IBNKCALL
#      CADR     E/JOBWAK
#      ...
#      RELINT      # RETURNS HERE
#                  # IF YOU DID AND INHINT.
#
#      BANK     33
#      SETLOC   E/PROG
#      BANK
#
#      COUNT*   $$/P07
#
E/JOBWAK      TC      JOBWAKE      # ARRIVE IWITH ADRES IN A.
              CS      BIT11
              NDX     LOCCTR
              ADS     LOC      # KNOCK FIXED MEMORY BIT OUT OF ADRES.
              TC      RUPTREG3  # RETURN
```

1412THE

IMU_PERFORMANCE_TEST_2

```
# NAME --          IMU PERFORMANCE TESTS 2
#
# DATE --          MARCH 20, 1967
#
# BY --            SYSTEM TEST GROUP 864-6900 EXT. 1274
#
# MODNO. --        ZERO
#
# FUNCTIONAL DESCRIPTION
#
# POSITIONING ROUTINES FOR THE IMU PERFORMANCE TESTS AS WELL AS SOME OF
# THE TESTS THEMSELVES.  FOR A DESCRIPTION OF THESE SUBROUTINES AND THE
# OPERATING PROCEDURES (TYPICALLY) SEE STG MEMO 685.  THEORETICAL REF. E-1973

          BANK      33
          SETLOC    IMU2
          BANK
          EBANK=     POSITON
          COUNT*    $$/P07

REDO      TC        NEWMODEX
          MM        07

GEOIMUTT  TC        IMUZERR
IMUBACK   CA        ZERO
          TS        NDXCTR
          TS        TORQNDX
          TS        TORQNDX +1
          TS        OVFLOWCK

NBPOSPL   CA        DEC17
          TS        ZERONDX
          CA        XNBADR
          TC        ZEROING
          CA        HALF
          TS        XNB

GUESS     TC        INTERPRET
LATAZCHK  DLOAD     SL2
          STODL     DSPTEM1 +1
          RTB       EXIT
          XCH       MPAC
          TS        DSPTEM1
          CAF       VN0641
          TC        BANKCALL
          CADR      GOFLASH
          TC        ENDTEST1
          TC        +2
          TC        -5
```

1412THE

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

[illegible]

	EXTEND		
	BZMF	+2	
	TC	ENDOFJOB	
	CA	FIVE	
	TS	RESULTCT	
	TC	CHECKG	
	CCS	DATAPL	+1
	TC	+4	
	TC	CCSHOLE	
	CS	DATAPL	+4
	TS	DATAPL	+4
	EXTEND		
	DCS	DATAPL	
	DAS	DATAPL	+4
	TC	INTPRET	
	DLOAD	DSU	
		DATAPL	+6
		DATAPL	+2
	BPL	CALL	
		AINGOTN	
		OVERFFIX	
AINGOTN	PDDL	DDV	
		DATAPL	+4
	DMPR	RTB	
		DEC585	# DEC585 HAS BEEN REDEFINED FOR LEM
		SGNAGREE	
	STORE	DSPTM2	
	EXIT		
	CCS	NDXCTR	
	TC	COALIGN	# TAKE PLATFORM OUT OF GIMBAL LOCK
	TC	SHOW	
VERTDRFT	CA	3990DEC	# ABOUT 1 HOUR VERTICAL DRIFT TEST
	TS	LENGTHOT	
	INDEX	POSITON	
	CS	SOUTHDR	-2
	TS	DRIFTT	
	CCS	PIPINDEX	# OFFSET PLATFORM TO MISS PIP DEAD-ZONES
	TCF	PON4	# Z-UP IN POS 4
PON2	CS	BIT5	# X-UP
	ADS	ERCOMP	+2
	CA	BIT5	
	ADS	ERCOMP	+4
	TCF	PON	
PON4	CS	BIT5	
	ADS	ERCOMP	+2
	CA	BIT5	
PON	ADS	ERCOMP	
	TC	EARTH*	



1				1
2		CA	ZERO	2
3		TS	ERVECTOR	3
4		TS	ERVECTOR +1	4
5	GUESS1	CAF	POSMAX	5
6		TS	TORQNDX	6
7		TS	TORQNDX +1	7
8		CA	CDUX	8
9		TS	LOSVEC	9
10		TC	ESTIMS	10
11	VALMIS	CA	DRIFT0	11
12		TS	DSPTM2 +1	12
13		CA	ZERO	13
14		TS	DSPTM2	14
15		TC	SHOW	15
16				16
17	ENDTEST1	TC	DOWNFLAG	17
18		ADRES	IMUSE	18
19		CS	ZERO	19
20		TC	NEWMODEA	20
21		TC	ENDEXT	21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1				1
2	OVERFFIX	DAD	DAD	2
3			DPPOS MAX	3
4			ONEDPP	4
5		RVQ		5
6				6
7	COAALIGN	EXTEND	# COARSE ALIGN SUBROUTINE	7
8		QXCH	ZERONDX	8
9		CA	ZERO	9
10		TS	THETAD	10
11		TS	THETAD +1	11
12		TS	THETAD +2	12
13		TC	BANKCALL	13
14		CADR	IMUCOARS	14
15	ALIGNCOA	TC	BANKCALL	15
16		CADR	IMUSTALL	16
17		TC	SOMERR2	17
18		TC	ZERONDX	18
19				19
20	IMUSLLLG	EXTEND		20
21		QXCH	ZERONDX	21
22		TC	ALIGNCOA	22
23				23
24	FINIMUDD	EXTEND		24
25		QXCH	ZERONDX	25
26		TC	BANKCALL	26
27		CADR	IMUFINE	27
28		TC	ALIGNCOA	28
29				29
30	IMUZERR	EXTEND		30
31		QXCH	ZERONDX	31
32		TC	BANKCALL	32
33		CADR	IMUZERO	33
34		TC	ALIGNCOA	34
35				35
36	CHECKG	EXTEND	# PIP PULSE CATCHING ROUTINE	36
37		QXCH	QPLACE	37
38		TC	+6	38
39	CHECKG1	RELINT		39
40		CA	NEWJOB	40
41		EXTEND		41
42		BZMF	+6	42
43		TC	CHANG1	43
44		INHINT		44
45		INDEX	PIPINDEX	45
46		CS	PIPAX	46
47		TS	ZERONDX	47
48		INHINT		48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60



1				1
2		INDEX	PIPINDEX	2
3		CA	PIPAX	3
4		AD	ZERONDX	4
5		EXTEND		5
6		BZF	CHECKG1	6
7		INDEX	PIPINDEX	7
8		CA	PIPAX	8
9		INDEX	RESULTCT	9
10		TS	DATAPL	10
11		TC	FINETIME	11
12		INDEX	RESULTCT	12
13		TS	DATAPL +1	13
14		INDEX	RESULTCT	14
15		LXCH	DATAPL +2	15
16		RELINT		16
17	ENDCHKG	TC	QPLACE	17
18				18
19	ZEROING	TS	L	19
20		TCF	+2	20
21	ZEROING1	TS	ZERONDX	21
22		CAF	ZERO	22
23		INDEX	L	23
24		TS	0	24
25		INCR	L	25
26		CCS	ZERONDX	26
27		TCF	ZEROING1	27
28		TC	Q	28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60



1				1
2	ERTHRVSE	DLOAD	PDDL	2
3			SCHZEROS	3
4			# PD24 = (SIN	4
5			-COS	5
6			0) (OMEG/MS)	6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

# INSTRUCTION PERFORMANCE TEST #2			PAGE 501	
1				1
2		QXCH	QPLACE	2
3	SHOW1	CA	POSITON	3
4		TS	DSPTM2 +2	5
5		CA	VB06N98	6
6		TC	BANKCALL	7
7		CADR	GOFLASH	9
8		TC	ENDTEST1 # V34	10
9		TC	QPLACE # V33	11
10		TCF	SHOW1	12
11				13
12	3990DEC	DEC	3990	14
13	VB06N98	VN	0698	15
14	VN0641	VN	0641	16
15	DEC17	=	ND1	17
16	DEC58	DEC	58	18
17	OGCPL	ECADR	OGC	19
18	1SECX	=	1SEC	20
19	XNBADR	GENADR	XNB	21
20	XSMADR	GENADR	XSM	22
21		BLOCK	2	23
22		COUNT*	\$\$/P07	24
23	FINETIME	INHINT	# RETURNS WITH INTERRUPT INHIBITED	25
24		EXTEND		26
25		READ	LOSCALAR	27
26		TS	L	28
27		EXTEND		29
28		RXOR	LOSCALAR	30
29		EXTEND		31
30		BZF	+4	32
31		EXTEND		33
32		READ	LOSCALAR	34
33		TS	L	35
34	+4	CS	POSMAX	36
35		AD	L	37
36		EXTEND		38
37		BZF	FINETIME +1	39
38		EXTEND		40
39		READ	HISCALAR	41
40		TC Q		42
41				43
42				44
43				45
44				46
45				47
46				48
47				49
48				50
49				51
50				52
51				53
52				54
53				55
54				56
55				57
56				58
57				59
58				60
59				61
60				62

IMU_PERFORMANCE_TESTS_4

```
# PROGRAM --      IMU PERFORMANCE TESTS 4
# DATE  --      NOV 15, 1966
# BY  --      GEORGE SCHMIDT IL7-146 EXT 1126
# MOD NO-ZERO
#
# FUNCTIONAL DESCRIPTION
#
# THIS SECTION CONSISTS OF THE FILTER FOR THE GYRO DRIFT TESTS.  NO COMPASS
# IS DONE IN LEM.  FOR A DESCRIPTION OF THE FILTER SEE E-1973.  THIS
# SECTION IS ENTERED FROM IMU 2.  IT RETURNS THERE AT END OF TEST.
#
# EARTH,OGC ZERO,ERTHRVSE
#
# NORMAL EXIT
#
# LENGTHOT GOES TO ZERO -- RETURN TO IMU PERF TESTS 2 CONTROL
#
# ALARMS
#
# 1600  OVERFLOW IN DRIFT TEST
# 1601  BAD IMU MODING IN ANY ROUTINE THAT USES IMUSTALL
#      OUTPUT
#
# FLASHING DISPLAY OF RESULTS -- CONTROLLED IN IMU PERF TESTS 2
#
# DEBRIS
#
# ALL CENTRALS -- ALL OF EBANK XSM
```


- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65
- 66
- 67
- 68
- 69
- 70
- 71
- 72
- 73
- 74
- 75
- 76
- 77
- 78
- 79
- 80

1412THE

1



1				1
2	ALLOOP	CA	OVFLOWCK	2
3		EXTEND		3
4		BZF	+2	4
5		TC	TASKOVER	5
6		CCS	ALTIM	6
7		CA	A	7
8		TS	ALTIMS	8
9		CS	A	9
10		TS	ALTIM	10
11		CS	ONE	11
12		AD	GEOCOMPS	12
13		EXTEND		13
14		BZF	+4	14
15		CA	LENGTHOT	15
16		EXTEND		16
17		BZMF	+5	17
18		CAE	1SECXT	18
19		TC	TWIDDLE	19
20		EBANK=	XSM	20
21		ADRES	ALLOOP	21
22		CAF	ZERO	22
23		XCH	PIPAX	23
24		TS	DELVX	24
25		CAF	ZERO	25
26		XCH	PIPAY	26
27		TS	DELVY	27
28		CAF	ZERO	28
29		XCH	PIPAZ	29
30		TS	DELVZ	30
31	SPECSTS	CAF	PRI020	31
32		TC	FINDVAC	32
33		EBANK=	XSM	33
34		2CADR	ALFLT	34
35			# START THE JOB	35
36		TC	TASKOVER	36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	# INCREMENT CHANGE FACTS			1
2	ALFLT	CCS	GEOCOMPS	2
3		TC	+2	3
4		TC	NORMLOP	5
5		TC	BANKCALL	6
6		CADR	1/PIPA	7
7	NORMLOP	TC	INTPRET	9
8		DLOAD		10
9			INTVAL	11
10		STOVL	S1	13
11			DELVX	14
12		VXM	VSL1	15
13			XSM	17
14		DLOAD	DCOMP	18
15			MPAC +3	19
16		STODL	DPIPAY	21
17			MPAC +5	22
18		STORE	DPIPAZ	23
19				25
20		SETPD	AXT,1	26
21			0	27
22				28
23		SLOAD	8D	29
24			DCOMP	30
25			GEOCOMPS	31
26		BMN		33
27	ALCGKK	SLOAD	PERFERAS	34
28			BMN	35
29				36
30	ALKCG	AXT,2	ALTIMS	37
31			ALFLT3	38
32			LXA,1	39
33			# LOADS SLOPES AND TIME CONSTANTS AT RQST	40
34	ALKCG2	DLOAD*	12D	41
35			ALX1S	42
36			INCR,1	43
37				44
38		DEC	ALFDK +144D,1	45
39		STORE	-2	46
40		TIX,2	ALDK +10D,2	47
41			SXA,1	48
42	ALFLT3	AXT,1	ALKCG2	49
43			ALX1S	50
44	DEMLP	DLOAD*		51
45			8D	52
46			DMP	53
47			DPIPAY +8D,1	54
48			PIPASC	55
49		SLR	BDSU*	56
50			9D	57
51			INTY +8D,1	58
52		STORE	INTY +8D,1	59
53		PDDL	DMP*	60
54			VELSC	61
55				62
56				63
57				64
58				65
59				66
60				67

1					1
2		VLAUN	+8D,1		2
3		SL2R			3
4		DSU	STADR		4
5		STORE	DELM	+8D,1	5
6		STORE	DELM	+10D,1	6
7		TIX,1	AXT,2		7
8			DELM		8
9			MLP		9
10	ALILP	DLOAD*	DMPR*		10
11			ALK	+4,2	11
12			ALDK	+4,2	12
13		STORE	ALK	+4,2	13
14		TIX,2	AXT,2		14
15			ALILP		15
16			8D		16
17	ALKLP	LXC,1	SXA,1		17
18			CMPX1		18
19			CMPX1		19
20		DLOAD*	DMPR*		20
21			ALK	+1,1	21
22			DELM	+8D,2	22
23		DAD*			23
24			INTY	+8D,2	24
25		STORE	INTY	+8D,2	25
26		DLOAD*	DAD*		26
27			ALK	+12D,2	27
28			ALDK	+12D,2	28
29		STORE	ALK	+12D,2	29
30		DMPR*	DAD*		30
31			DELM	+8D,2	31
32			INTY	+16D,2	32
33		STORE	INTY	+16D,2	33
34		DLOAD*	DMP*		34
35			ALSK	+1,1	35
36			DELM	+8D,2	36
37		SL1R	DAD*		37
38			VLAUN	+8D,2	38
39		STORE	VLAUN	+8D,2	39
40		TIX,2	AXT,1		40
41			ALKLP		41
42			8D		42
43					43
44	LOOSE	DLOAD*	PDDL*		44
45			ACCWD	+8D,1	45
46			VLAUN	+8D,1	46
47		PDDL*	VDEF		47
48			POSNV	+8D,1	48
49		MXV	VSL1		49
50			TRANSM1		50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1					1
2		DLOAD			2
3			MPAC		3
4		STORE	POSNV	+8D,1	4
5		DLOAD			5
6			MPAC	+3	6
7		STORE	VLAUN	+8D,1	7
8		DLOAD			8
9			MPAC	+5	9
10		STORE	ACCWD	+8D,1	10
11		TIX,1			11
12			LOOSE		12
13					13
14		AXT,2	AXT,1	# EVALUATE SINES AND COSINES	14
15			6		15
16			2		16
17	BOOP	DLOAD*	DMPR		17
18			ANGX	+2,1	18
19			GEORGEJ		19
20		SR2R			20
21		PUSH	SIN		21
22		SL3R	XAD,1		22
23			X1		23
24		STORE	16D,2		24
25		DLOAD			25
26		COS			26
27		STORE	22D,2	# COSINES	27
28		TIX,2			28
29			BOOP		29
30					30
31	PERFERAS	EXIT			31
32		CA	EBANK7		32
33		TS	EBANK		33
34		EBANK=	ATIGINC		34
35		TC	ATIGINC	# GOTO ERASABLE TO CALCULATE ONLY TO RETN	35
36					36
37	#	CAUTION			37
38	#				38
39	#	THE ERASABLE PROGRAM THAT DOES THE CALCULATIONS MUST BE LOADED			39
40	#	BEFORE ANY ATTEMPT IS MAKE TO RUN THE IMU PERFORMANCE TEST			40
41					41
42		EBANK=	AZIMUTH		42
43		CCS	LENGTHOT		43
44		TC	SLEEPIE		44
45		CCS	TORQNDX		45
46		TCF	+2		46
47		TC	SETUPER1		47
48		CA	CDUX		48
49		TS	LOSVEC	+1 # FOR TROUBLESHOOTING VD POSNS 2\$4	49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1				1
2	SETUPER1	TC	INTPRET	2
3		DLOAD	PDDL	3
4			# ANGLES FROM DRIFT TEST ONLY	4
5			ANGZ	5
6			ANGY	6
7		PDDL	VDEF	7
8			ANGX	8
9		VCOMP	VXSC	9
10			GEORGEJ	10
11		MXV	VSR1	11
12			XSM	12
13		STORE	OGC	13
14		EXIT		14
15		CA	OGCPL	15
16		TC	BANKCALL	16
17		CADR	IMUPULSE	17
18		TC	IMUSLLL	18
19	GEOSTR4	CCS	TORQNDX	19
20		TC	VALMIS	20
21		TC	INTPRET	21
22		CALL		22
23			ERTHRVSE	23
24		EXIT		24
25		TC	TORQUE	25
26				26
27	SLEEPIE	TS	LENGTHOT	27
28		CCS	TORQNDX	28
29		TC	EARTH*	29
30		TC	ENDOFJOB	30
31				31
32	SOMEERRR	CA	EBANK5	32
33		TS	EBANK	33
34		CA	ONE	34
35		TS	OVFLOWCK	35
36		TC	ALARM	36
37		OCT	1600	37
38		TC	ENDTEST1	38
39	SOMERR2	CAF	OCT1601	39
40		TC	VARALARM	40
41		TC	DOWNFLAG	41
42		ADRES	IMUSE	42
43		TC	ENDOFJOB	43
44				44
45	OCT1601	OCT	01601	45
46	DEC585	OCT	06200	46
47	SCHZEROS	2DEC	.00000000	47
48			# 3200 B+14 ORDER IS IMPORTANT	48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60


```
1 # PROGRAM NAME -- KEYBOARD AND DISPLAY PROGRAM
2 # MOD NO -- 4          DATE -- 27 APRIL 1967          ASSEMBLY -- PINDANCE REV 18
3
4 # MOD BY -- FILENE
5 # LOG SECTION -- PINBALL GAME BUTTONS AND LIGHTS
6 #
7 # FUNCTIONAL DESCRIPTION
8 #
9 # THE KEYBOARD AND DISPLAY SYSTEM PROGRAM OPERATES UNDER EXECUTIVE
10 # CONTROL AND PROCESSES INFORMATION EXCHANGED BETWEEN THE AGC AND THE
11 # COMPUTER OPERATOR.  THE INPUTS TO THE PROGRAM ARE FROM THE KEYBOARD,
12 # FROM INTERNAL PROGRAM, AND FROM THE UPLINK.
13 #
14 # THE LANGUAGE OF COMMUNICATION WITH THE PROGRAM IS A PAIR OF WORDS
15 # KNOWN AS VERB AND NOUN.  EACH OF THESE IS REPRESENTED BY A 2 CHARACTER
16 # DECIMAL NUMBER.  THE VERB CODE INDICATES WHAT ACTION IS TO BE TAKEN, THE
17 # NOUN CODE INDICATES TO WHAT THIS ACTION IS APPLIED.  NOUNS USUALLY
18 # REFER TO A GROUP OF ERASABLE REGISTERS.
19 #
20 # VERBS ARE GROUPED INTO DISPLAYS, LOADS, MONITORS (DISPLAYS THAT ARE
21 # UPDATED ONCE PER SECOND), SPECIAL FUNCTIONS, AND EXTENDED VERBS (THESE
22 # ARE OUTSIDE OF THE DOMAIN OF PINBALL AND CAN BE FOUND UNDER LOG SECTION
23 # 'EXTENDED VERBS').
24 #
25 # A LIST OF VERBS AND NOUNS IS GIVEN IN LOG SECTION 'ASSEMBLY AND
26 # OPERATION INFORMATION'.
27 #
28 #
29 # CALLING SEQUENCES --
30 #
31 # KEYBOARD:
32 # EACH DEPRESSION OF A KEYBOARD BUTTON ACTIVATES AN INTERRUPT KEYRUPT1
33 # AND PLACES THE 5 BIT KEY CODE INTO CHANNEL 15.  KEYRUPT1 PLACES THE KEY
34 # CODE INTO MPAC, ENTERS AN EXECUTIVE REQUEST FOR THE KEYBOARD AND DISPLAY
35 # PROGRAM (AT 'CHARIN'), AND EXECUTES A RESUME.
36 #
37 # UPLINK:
38 # EACH WORD RECEIVED BY THE UPLINK ACTIVATES INTERRUPT UPRUPT, WHICH
39 # PLACES THE 5 BIT KEY CODE INTO MPAC, ENTERS AN EXECUTIVE REQUEST FOR THE
40 # KEYBOARD AND DISPLAY PROGRAM (AT 'CHARIN') AND EXECUTES A RESUME.
41 #
42 # INTERNAL PROGRAMS:
43 # INTERNAL PROGRAMS CALL PINBALL AT 'NVSUB' WITH THE DESIRED VERB/NOUN
44 # CODE IN A (LOW 7 BITS FOR NOUN, NEXT 7 BITS FOR VERB).  DETAILS
45 # DESCRIBED ON REMARKS CARDS JUST BEFORE 'NVSUB' AND 'NVSUBWAIT' (SEE
46 # SYMBOL TABLE FOR PAGE NUMBERS).
47 #
48 # NORMAL EXIT MODES --
49 #
50 # IF PINBALL WAS CALLED BY EXTERNAL ACTION, THERE ARE FOUR EXITS:
51 # 1) ALL BUT (2), (3), AND (4) EXIT DIRECTLY TO ENDOFJOB.
52 # 2) EXTENDED VERBS GO TO THE EXTENDED VERB FAN AS PART OF THE
```



```
1 #
2 # PINBALL EXECUTIVE JOB WITH PRIORITY 30000. IT IS THE
3 # RESPONSIBILITY OF THE EXTENDED VERB CALLED TO EVENTUALLY
4 # CHANGE PRIORITY (IF NECESSARY) AND DO AN ENDOFJOB.
5 # ALSO PINBALL IS A NOVAC JOB. EBANK SET FOR COMMON.
6 # 3) VERB 37. CHANGE OF PROGRAM (MAJOR MODE) CALLS 'V37' IN THE
7 # SERVICE ROUTINES AS PART OF THE PINBALL EXEC JOB WITH PRIORITY
8 # 30000. THE NEW PROGRAM CODE (MAJOR MODE) IS LEFT IN A.
9 # 4) KEY RELEASE BUTTON CALLS 'PINBRNCH' IN THE DISPLAY INTERFACE
10 # ROUTINES AS PART OF THE PINBALL EXEC JOB WITH PRIORITY 30000 IF
11 # THE KEY RELEASE LIGHT IS OFF AND 'CADRSTOR' IS NOT +0.
12 #
```

```
13 # IF PINBALL WAS CALLED BY INTERNAL PROGRAMS, EXIT FROM PINBALL IS BACK
14 # TO CALLING ROUTINE. DETAILS DESCRIBED IN REMARKS CARDS JUST BEFORE
15 # 'NVSUB' AND 'NVSUBWAIT' (SEE SYMBOL TABLE FOR PAGE NUMBERS).
```

```
16 #
17 # ALARM OR ABORT EXIT MODES --
```

```
18 #
19 # EXTERNAL INITIATION:
20 # IF SOME IMPROPER SEQUENCE OF KEY CODES IS DETECTED, THE OPERATOR
21 # ERROR LIGHT IS TURNED ON AND EXIT IS TO 'ENDOFJOB'.
```

```
22 #
23 # INTERNAL PROGRAM INITIATION:
24 # IF AN ILLEGAL V/N COMBINATION IS ATTEMPTED, AN ABORT IS CAUSED
25 # (WITH OCTAL 01501).
26 # IF A SECOND ATTEMPT IS MADE TO GO TO SLEEP IN PINBALL, AN ABORT IS
27 # CAUSED (WITH OCTAL 01206). THERE ARE TWO WAYS TO GO TO SLEEP IN PINBALL:
```

- ```
28 # 1) ENDIDLE OR DATAWAIT.
29 # 2) NVSUBWAIT, PRENVBSY, OR NVSUBUSY.
30 #
```

```
31 # CONDITIONS LEADING TO THE ABOVE ARE DESCRIBED IN FORTHCOMING MIT/IL
32 # E-REPORT DESCRIBING KEYBOARD AND DISPLAY OPERATION FOR 278.
```

```
33 #
34 # OUTPUT --
```

```
35 #
36 # INFORMATION TO BE SENT TO THE DISPLAY PANEL IS LEFT IN THE 'DSPTAB'
37 # BUFFERS REGISTERS (UNDER EXEC CONTROL). 'DSPOUT' (A PART OF T4RUPT)
38 # HANDLES THE PLACING OF THE 'DSPTAB' INFORMATION INTO OUTPUT CHANNEL 10
39 # IN INTERRUPT.
```

```
40 #
41 # ERASABLE INITIALIZATION --
```

```
42 #
43 # FRESH START AND RESTART INITIALIZE THE NECESSARY E REGISTERS FOR
44 # PINBALL IN 'STARTSUB'. REGISTERS ARE: DSPTAB BUFFER, CADRSTOR,
45 # REQRET, CLPASS, DSPLOCK, MONSAVE, MONSAVE1, VERBREG, NOUNREG, DSPLIST,
46 # DSPCOUNT, NOUT.
```

```
47 #
48 # A COMPLETE LIST OF ALL THE ERASABLES (BOTH RESERVED AND TEMPORARIES) FOR
```

# PINBALL IS GIVEN BELOW.

# THE FOLLOWING ARE OF GENERAL INTEREST --

# REMARKS CARDS PRECEDE THE REFERENCED SYMBOL DEFINITION. SEE SYMBOL  
# TABLE TO FIND APPROPRIATE PACE NUMBERS.

# NVSUB CALLING POINT FOR INTERNAL USE OF PINBALL.  
# OF RELATED INTEREST NVSBWAIT  
# NVSUBUSY  
# PRENVBSY

# ENDIDLE ROUTINE FOR INTERNAL PROGRAMS WISHING TO TO SLEEP WHILE  
# AWAITING OPERATORS RESPONSE.

# DSPMM ROUTINE BY WHICH AN INTERNAL PROGRAM MAY DISPLAY A DECIMAL  
# PROGRAM CODE (MAJOR MODE) IN THE PROGRAM (MAJOR MODE) LIGHTS.  
# (DSPMM DOES NOT DISPLAY DIRECTLY BUT ENTERS EXEC REQUEST  
# FOR DSPMMJB WITH PRIO 30000 AND RETURNS TO CALLER.)

# BLANKSUB ROUTINE BY WHICH AN INTERNAL PROGRAM MAY BLANK ANY  
# COMBINATION OF THE DISPLAY REGISTERS R1, R2, R3.

# JAMTERM ROUTINE BY WHICH AN INTERNAL PROGRAM MAY PERFORM THE  
# JAMPROC TERMINATE (V 34) OR PROCEED (V 33) FUNCTION.

# MONITOR VERBS FOR PERIODIC (1 PER SEC) DISPLAY.

# PLEASE PERFORM, PLEASE MARK SITUATIONS  
# REMARKS DESCRIBING HOW AN INTERNAL ROUTINE SHOULD HANDLE  
# THESE SITUATIONS CAN BE FOUND JUST BEFORE 'NVSUB' (SEE  
# SYMBOL TABLE FOR PAGE NUMBER).

# THE NOUN TABLE FORMAT IS DESCRIBED ON A PAGE OF REMARKS CARDS JUST  
# BEFORE 'DSPABC' (SEE SYMBOL TABLE FOR PAGE NUMBER).

# THE NOUN TABLES THEMSELVES ARE FOUND IN LOG SECTION 'PINBALL NOUN  
# TABLES'.

# FOR FURTHER DETAILS ABOUT OPERATION OF THE KEYBOARD AND DISPLAY SYSTEM  
# PROGRAM, SEE THE MISSION PLAN AND/OR MIT/IL E-2129  
# DESCRIBING KEYBOARD AND DISPLAY OPERATION FOR 278.

# THE FOLLOWING QUOTATION IS PROVIDED THROUGH THE COURTESY OF THE AUTHORS.

# "IT WILL BE PROVED TO THY FACE THAT THOU HAST MEN ABOUT THEE THAT  
# USUALLY TALK OF A NOUN AND A VERB, AND SUCH ABOMINABLE WORDS AS NO

# CHRISTIAN EAR CAN ENDURE TO HEAR."

# HENRY 6, ACT 2, SCENE 4

# THE FOLLOWING ASSIGNMENTS FOR PINBALL ARE MADE ELSEWHERE

|            |       |        |                                               |
|------------|-------|--------|-----------------------------------------------|
| #          |       |        | # DISPLAY POSITION INDICATOR                  |
| # DSPCOUNT | ERASE |        | # +DEC, -DEC, OCT INDICATOR                   |
| # DECBRNCH | ERASE |        |                                               |
| # VERBREG  | ERASE |        | # VERB CODE                                   |
| # NOUNREG  | ERASE |        | # NOUN CODE                                   |
| # XREG     | ERASE |        | # R1 INPUT BUFFER                             |
| # YREG     | ERASE |        | # R2 INPUT BUFFER                             |
| # ZREG     | ERASE |        | # R3 INPUT BUFFER                             |
| # XREGLP   | ERASE |        | # LO PART OF XREG (FOR DEC CONV ONLY)         |
| # YREGLP   | ERASE |        | # LO PART OF YREG (FOR DEC CONV ONLY)         |
| # HITEMOUT | =     | YREGLP | # TEMP FOR DISPLAY OF HRS, MIN, SEC           |
| #          |       |        | # MUST = LOTEMOUT-1.                          |
| # ZREGLP   | ERASE |        | # LO PART OF ZREG (FOR DEC CONV ONLY)         |
| # LOTEMOUT | =     | ZREGLP | # TEMP FOR DISPLAY OF HRS, MIN, SEC           |
| #          |       |        | # MUST = HITEMOUT+1.                          |
| # MODREG   | ERASE |        | # MODE CODE                                   |
| # DSPLOCK  | ERASE |        | # KEYBOARD/SUBROUTINE CALL INTERLOCK          |
| # REQRET   | ERASE |        | # RETURN REGISTER FOR LOAD                    |
| # LOADSTAT | ERASE |        | # STATUS INDICATOR FOR LOADTST                |
| # CLPASS   | ERASE |        | # PASS INDICATOR FOR CLEAR                    |
| # NOUT     | ERASE |        | # ACTIVITY COUNTER FOR DSPTAB                 |
| # NOUNCADR | ERASE |        | # MACHINE CADR FOR NOUN                       |
| # MONSAVE  | ERASE |        | # N/V CODE FOR MONITOR. (= MONSAVE1-1)        |
| # MONSAVE1 | ERASE |        | # NOUNCADR FOR MONITOR (MATBS) = MONSAVE+1    |
| # MONSAVE2 | ERASE |        | # NVMONOPT OPTIONS                            |
| # DSPTAB   | ERASE | +13D   | # 0-10, DISPLAY PANEL BUFFER, 11-13, C RELAYS |
| # CADRSTOR | ERASE |        | # ENDIDLE STORAGE                             |
| # NVQTEM   | ERASE |        | # NVSUB STORAGE FOR CALLING ADDRESS           |
| #          |       |        | # MUST = NVBNKTEM-1.                          |
| # NVBNKTEM | ERASE |        | # NVSUB STORAGE FOR CALLING BANK              |
| #          |       |        | # MUST = NVQTEM+1                             |
| # VERBSAVE | ERASE |        | # NEEDED FOR RECYCLE                          |
| # DSPLIST  | ERASE |        | # WAITING REG FOR DSP SYST INTERNAL USE       |
| # EXTVBACT | ERASE |        | # EXTENDED VERB ACTIVITY INTERLOCK            |
| # DSPTM1   | ERASE |        | # BUFFER STORAGE AREA 1 (MOSTLY FOR TIME)     |
| # DSPTM2   | ERASE |        | # BUFFER STORAGE AREA 2 (MOSTLY FOR DEG)      |

# END OF ERASABLES RESERVED FOR PINBALL EXECUTIVE ACTION

#

# TEMPORARIES FOR PINBALL EXECUTIVE ACTION

```
1
2
3 # DSEXIT = INTB15+ # RETURN FOR DSPIN
4 # EXITEM = INTB15+ # RETURN FOR SCALE FACTOR ROUTINE SELECT
5 # BLANKRET = INTB15+ # RETURN FOR 2BLANK
6
7 # WRDRET = INTBIT15 # RETURN FOR 5BLANK.
8 # WDRET = INTBIT15 # RETURN FOR DSPWD
9 # DECRET = INTBIT15 # RETURN FOR PUTCOM(DEC LOAD)
10 # 21/22REG = INTBIT15 # TEMP FOR CHARIN
11
12 # UPDATRET = POLISH # RETURN FOR UPDATNN, UPDATVB
13 # CHAR = POLISH # TEMP FOR CHARIN
14 # ERCNT = POLISH # COUNTER FOR ERROR LIGHT RESET
15 # DECOUNT = POLISH # COUNTER FOR SCALING AND DISPLAY (DEC)
16
17 # SGNON = VBUF # TEMP FOR +,- ON
18 # NOUNTEM = VBUF # COUNTER FOR MIXNOUN FETCH
19 # DISTEM = VBUF # COUNTER FOR OCTAL DISPLAY VERB
20 # DECTEM = VBUF # COUNTER FOR FETCH (DEC DISPLAY VERBS)
21
22 # SGNOFF = VBUF +1 # TEMP FOR +,- ON
23 # NVTEMP = VBUF +1 # TEMP FOR NVSUB
24 # SFTEMP1 = VBUF +1 # STORAGE FOR SF CONST HI PART (=SFTEMP2-1)
25 # HITEMIN = VBUF +1 # TEMP FOR LOAD OF HRS, MIN, SEC
26 # = # MUST = LOTEMIN-1.
27 # CODE = VBUF +2 # FOR DSPIN
28 # SFTEMP2 = VBUF +2 # STORAGE FOR SF CONST LO PART (=SFTEMP1+1)
29 # LOTEMIN = VBUF +2 # TEMP FOR LOAD OF HRS, MIN, SEC
30 # = # MUST = HITEMIN+1
31 # MIXTEMP = VBUF +3 # FOR MIXNOUN DATA
32 # SIGNRET = VBUF +3 # RETURN FOR +,- ON
33 # ALSO MIXTEMP+1 = VBUF+4, MIXTEMP+2 = VBUF+5.
34
35 # ENTRET = DOTINC # EXIT FROM ENTER
36
37 # WDCNT = DOTRET # CHAR COUNTER FOR DSPWD
38 # INREL = DOTRET # INPUT BUFFER SELECTOR (X, Y, Z, REG)
39
40 # DSPMMTEM = MATINC # DSPCOUNT SAVE FOR DSPMM
41 # MIXBR = MATINC # INDICATOR FOR MIXED OR NORMAL NOUN
42
43 # TEM1 ERASE
44 # DSREL = TEM1 # REL ADDRESS FOR DSPIN
45
46 # TEM2 ERASE
47 # DSMAG = TEM2 # MAGNITUDE STORE FOR DSPIN
48 # IDADDTEM = TEM2 # MIXNOUN INDIRECT ADDRESS STORAGE
49 # TEM3 ERASE
50 # COUNT = TEM3 # FOR DSPIN
51
52
53
54
55
56
57
58
59
60
```

```
1 # TEM4 ERASE # EXEC TEMP
2 # LSTPTR = TEM4 # LIST POINTER FOR GRABUSY
3
4 # RELRET = TEM4 # RETURN FOR RELDSP
5 # FREERET = TEM4 # RETURN FOR FREEDSP
6 # DSPWDRET = TEM4 # RETURN FOR DSPSIGN
7 # SEPSCRET = TEM4 # RETURN FOR SEPSEC
8 # SEPMNRET = TEM4 # RETURN FOR SEPMIN
9
10 # TEM5 ERASE # EXEC TEMP
11 # NOUNADD = TEM5 # TEMP STORAGE FOR NOUN ADDRESS
12
13 # NNADTEM ERASE # TEMP FOR NOUN ADDRESS TABLE ENTRY
14 # NNTYPTTEM ERASE # TEMP FOR NOUN TYPE TABLE ENTRY
15 # IDAD1TEM ERASE # TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)
16 # # MUST = IDAD2TEM-1, = IDAD3TEM-2.
17 # IDAD2TEM ERASE # TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)
18 # # MUST = IDAD1TEM+1, IDAD3TEM-1.
19 # IDAD3TEM ERASE # TEMP FOR INDIR ADDRESS TABLE ENTRY (MIXNN)
20 # # MUST = IDAD1TEM+2, IDAD2TEM+1.
21 # RUTMXTEM ERASE # TEMP FOR SF ROUT TABLE ENTRY (MIXNN ONLY)
22 #
23 # END OF TEMPORARIES FOR PINBALL EXECUTIVE ACTION.
24 #
25 # ADDITIONAL TEMPORARIES FOR PINBALL EXECUTIVE ACTION
26 #
27 # MPAC, THRU MPAC +6
28 # BUF, +1, +2
29 # BUF2, +1, +2
30 # MPTEMP
31 # ADDRWD
32 #
33 # END OF ADDITIONAL TEMPS FOR PINBALL EXEC ACTION
34 #
35 # RESERVED FOR PINBALL INTERRUPT ACTION
36 #
37 # DSPCNT ERASE # COUNTER FOR DSPOUT
38 # UPLOCK ERASE # BIT1 = UPLINK INTERLOCK (ACTIVATED BY
39 # # RECEPTION OF A BAD MESSAGE IN UPLINK)
40 #
41 # END OF ERASABLES RESERVED FOR PINBALL INTERRUPT ACTION
42 #
43 # TEMPORARIES FOR PINBALL INTERRUPT ACTION
44 #
45 # KEYTEMP1 = WAITEXIT # TEMP FOR KEYRUPT, UPRUPT
46 # DSRUPTTEM = WAITEXIT # TEMP FOR DSPOUT
47 # KEYTEMP2 = RUPTAGN # TEMP FOR KEYRUPT, UPRUPT
48 #
49 # END OF TEMPORARIES FOR PINBALL INTERRUPT ACTION
```

# THE INPUT CODES ASSUMED FOR THE KEYBOARD ARE,

# 0 10000

# 1 00001

# 9 01001

# VERB 10001

# ERROR RES 10010

# KEY RLSE 11001

# + 11010

# - 11011

# ENTER 11100

# CLEAR 11110

# NOUN 11111

#

# (2003 RSB -- THE PROCEED KEY HAS NO KEYCODE; IT IS READ BY AN ALTERNATE MECHANISM.)

#

# OUTPUT FORMAT FOR DISPLAY PANEL. SET OUTO TO AAAABCCCCCDDDDDD.

# A'S SELECTS A RELAYWORD. THIS DETERMINES WHICH PAIR OF CHARACTERS ARE

# ENERGIZED.

# B FOR SPECIAL RELAYS SUCH AS SIGNS ETC.

# C'S 5 BIT RELAY CODE FOR LEFT CHAR OF PAIR SELECTED BY RELAYWORD.

# D'S 5 BIT RELAY CODE FOR RIGHT CHAR OF PAIR SELECTED BY RELAYWORD.

#

# THE PANEL APPEARS AS FOLLOWS,

# MD1 MD2 (MAJOR MODE)

# VD1 VD2 (VERB) ND1 ND2 (NOUN)

# R1D1 R1D2 R1D3 R1D4 R1D5 (R1)

# R2D1 R2D2 R2D3 R2D4 R2D5 (R2)

# R3D1 R3D2 R3D3 R3D4 R3D5 (R3)

#

# EACH OF THESE IS GIVEN A DSPCOUNT NUMBER FOR USE WITHIN COMPUTATION ONLY

#

# MD1 25 R2D1 11 ALL ARE OCTAL

# MD2 24 R2D2 10

# VD1 23 R2D3 7

# VD2 22 R2D4 6

# ND1 21 R2D5 5

# ND2 20 R3D1 4

# R1D1 16 R3D2 3

# R1D2 15 R3D3 2

# R1D3 14 R3D4 1

# R1D4 13 R3D5 0

# R1D5 12

#

# THERE IS AN 11-REGISTER TABLE (DSPTAB) FOR THE DISPLAY PANEL.

#

# DSPTAB RELAYWD BIT11 BITS 10-6 BITS 5-1

# RELADD

# 10 1011 MD1 (25) MD2 (24)

# 9 1010 VD1 (23) VD2 (22)

# 8 1001 ND1 (21) ND2 (20)

# 7 1000 R1D1 (16)

1412THE



## # START OF EXECUTIVE SECTION OF PINBALL

BANK 40  
SETLOC PINBALL1  
BANK

CHARIN COUNT\* \$\$/PIN  
CAF ONE # BLOCK DISPLAY SYST  
XCH DSPLOCK # MAKE DSP SYST BUSY, BUT SAVE OLD  
TS 21/22REG # C(DSPLOCK) FOR ERROR LIGHT RESET.  
CCS CADRSTOR # ALL KEYS EXCEPT ER TURN ON KR LITE IF  
TC +2 # CADRSTOR IS FULL. THIS REMINDS OPERATOR  
TC CHARIN2 # TO RE-ESTABLISH A FLASHING DISPLAY  
CS ELRCODE1 # WHICH HE HAS OBSCURED WITH DISPLAYS OF  
AD MPAC # HIS OWN (SEE REMARKS PRECEDING ROUTINE  
EXTEND # VBRELDSP).

CHARIN2 TC RELDSPON  
XCH MPAC  
TS CHAR

INDEX A  
TC +1 # INPUT CODE FUNCTION  
TC CHARALRM # 0

TC NUM # 1  
TC NUM # 2  
TC NUM # 3  
TC NUM # 4  
TC NUM # 5  
TC NUM # 6

TC NUM # 7  
TC 89TEST # 10 8  
TC 89TEST # 11 9

TC CHARALRM # 12  
TC CHARALRM # 13  
TC CHARALRM # 14

TC CHARALRM # 15  
TC CHARALRM # 16  
TC CHARALRM # 17

TC NUM -2 # 20 0  
TC VERB # 21 VERB  
TC ERROR # 22 ERROR LIGHT RESET

TC CHARALRM # 23  
TC CHARALRM # 24  
TC CHARALRM # 25

TC CHARALRM # 26  
TC CHARALRM # 27  
TC CHARALRM # 30

TC VBRELDSP # 31 KEY RELEASE  
TC POSGN # 32 +



```
1
2 TC NEGSN # 33 -
3 TC ENTERJMP # 34 ENTER
4
5 TC CHARALRM # 35
6 TC CLEAR # 36 CLEAR
7 TC NOUN # 37 NOUN
8
9 ELRCODE1 OCT 22
10 ENTERJMP TC POSTJUMP
11
12 89TEST CCS DSPCOUNT
13 TC +4 # +
14 TC +3 # +0
15 TC ENDOFJOB # - BLOCK DATA IN IF DSPCOUNT IS - OR -0
16 TC ENDOFJOB # -0
17 CAF THREE
18 MASK DECBRNCH
19
20 CCS A
21 TC NUM # IF DECBRNCH IS +, 8 OR 9 OK
22 TC CHARALRM # IF DECBRNCH IS +0, REJECT 8 OR 9
```

```
23 # NUM ASSEMBLES OCTAL 3 BITS AT A TIME. FOR DECIMAL IT CONVERTS INCOMING
24 # WORD AS A FRACTION, KEEPING RESULTS TO DP.
25 # OCTAL RESULTS ARE LEFT IN XREG, YREG, OR ZREG. HI PART OF DEC IN XREG,
26 # YREG, ZREG. THE LOW PARTS IN XREGLP, YREGLP, OR ZREGLP.
27 # DECBRNCH IS LEFT AT +0 FOR OCT, +1 FOR + DEC, +2 FOR - DEC.
28 # IF DSPCOUNT WAS LEFT -, NO MORE DATA IS ACCEPTED.
```

```
29
30 CAF ZERO
31 NUM TS CHAR
32 CCS DSPCOUNT
33 TC +4 # +
34 TC +3 # +0
35 TC +1 # -BLOCK DATA IN IF DSPCOUNT IS -
36 TC ENDOFJOB # -0
37
38 TC GETINREL
39 CCS CLPASS # IF CLPASS IS + OR +0, MAKE IT +0.
40 CAF ZERO
41 TS CLPASS
42 TC +1
43 INDEX CHAR
44 CAF RELTAB
45 MASK LOW5
46 TS CODE
47 CA DSPCOUNT
48 TS COUNT
49 TC DSPIN
50 CAF THREE
```

|          |        |           |                                              |
|----------|--------|-----------|----------------------------------------------|
|          | MASK   | DECBRNCH  |                                              |
|          | CCS    | A         | # +0, OCTAL. +1, + DEC. +2, - DEC.           |
|          | TC     | DECTOBIN  | # +                                          |
|          | INDEX  | INREL     | # +0 OCTAL                                   |
|          | XCH    | VERBREG   |                                              |
|          | TS     | CYL       |                                              |
|          | CS     | CYL       |                                              |
|          | CS     | CYL       |                                              |
|          | XCH    | CYL       |                                              |
|          | AD     | CHAR      |                                              |
|          | TC     | ENDNMTST  |                                              |
| DECTOBIN | INDEX  | INREL     |                                              |
|          | XCH    | VERBREG   |                                              |
|          | TS     | MPAC      | # SUM X 2EXP-14 IN MPAC                      |
|          | CAF    | ZERO      |                                              |
|          | TS     | MPAC +1   |                                              |
|          | CAF    | TEN       | # 10 X 2EXP-14                               |
|          | TC     | SHORTMP   | # 10SUM X 2EXP-28 IN MPAC, MPAC+1            |
|          | XCH    | MPAC +1   |                                              |
|          | AD     | CHAR      |                                              |
|          | TS     | MPAC +1   |                                              |
|          | TC     | ENDNMTST  | # NO OF                                      |
|          | ADS    | MPAC      | # OF MUST BE 5TH CHAR                        |
|          | TC     | DECEND    |                                              |
| ENDNMTST | INDEX  | INREL     |                                              |
|          | TS     | VERBREG   |                                              |
|          | CS     | DSPCOUNT  |                                              |
|          | INDEX  | INREL     |                                              |
|          | AD     | CRITCON   |                                              |
|          | EXTEND |           |                                              |
|          | BZF    | ENDNUM    | # -0, DSPCOUNT = CRITCON                     |
|          | TC     | MORNUM    | # - , DSPCOUNT G/ CRITCON                    |
| ENDNUM   | CAF    | THREE     |                                              |
|          | MASK   | DECBRNCH  |                                              |
|          | CCS    | A         |                                              |
|          | TC     | DECEND    |                                              |
| ENDALL   | CS     | DSPCOUNT  | # BLOCK NUMIN BY PLACING DSPCOUNT            |
|          | TC     | MORNUM +1 | # NEGATIVELY                                 |
| DECEND   | CS     | ONE       |                                              |
|          | AD     | INREL     |                                              |
|          | EXTEND |           |                                              |
|          | BZMF   | ENDALL    | # IF INREL=0,1 (VBREG,NNREG) LEAVE WHOLE     |
|          | TC     | DMP       | # IF INREL=2,3,4 (R1,R2,R3), CONVERT TO FRAC |
|          |        |           | # MULT SUM X 2EXP-28 IN MPAC, MPAC+1 BY      |
|          | ADRES  | DECON     | # 2EXP14/10EXP5, GIVES (SUM/10EXP5)X2EXP-14  |
|          | CAF    | THREE     | # IN MPAC, +1, +2.                           |
|          | MASK   | DECBRNCH  |                                              |
|          | INDEX  | A         |                                              |
|          | TC     | +0        |                                              |
|          | TC     | +DECSGN   |                                              |

```
1
2 EXTEND # - CASE
3 DCS MPAC +1
4 +DECSGN DXCH MPAC +1
5 XCH MPAC +2
6 INDEX INREL
7
8 TS XREGLP -2
9 XCH MPAC +1
10 INDEX INREL
11
12 MORNUM TS VERBREG
13 TC ENDALL
14 CCS DSPCOUNT # DECREMENT DSPCOUNT
15
16 TS DSPCOUNT
17 TC ENDOFJOB
18
19 CRITCON OCT 22 # (DEC 18)
20 OCT 20 # (DEC 16)
21 OCT 12 # (DEC 10)
22
23 OCT 5
24 OCT 0
25
26 DECON 2DEC 1 E-5 B14 # 2EXP14/10EXP5 = .16384 DEC
27
28 # GETINREL GETS PROPER DATA REG REL ADDRESS FOR CURRENT C(DSPCOUNT) AND
29 # PUTS IN INTO INREL. +0 VERBREG, 1 NOUNREG, 2 XREG, 3 YREG, 4 ZREG.
30
31 GETINREL INDEX DSPCOUNT
32 CAF INRELTAB
33 TS INREL # (A TEMP. REG)
34 TC Q
35
36 INRELTAB OCT 4 # R3D5 (DSPCOUNT = 0)
37 OCT 4 # R3D4 =(1)
38 OCT 4 # R3D3 =(2)
39 OCT 4 # R3D2 =(3)
40 OCT 4 # R3D1 =(4)
41 OCT 3 # R2D5 =(5)
42 OCT 3 # R2D4 =(6)
43 OCT 3 # R2D3 =(7)
44 OCT 3 # R2D2 =(8D)
45 OCT 3 # R2D1 =(9D)
46 OCT 2 # R1D5 =(10D)
47 OCT 2 # R1D4 =(11D)
48 OCT 2 # R1D3 =(12D)
49 OCT 2 # R1D2 =(13D)
50 OCT 2 # R1D1 =(14D)
51 TC CCSHOLE # NO DSPCOUNT NUMBER = 15D
52 OCT 1 # ND2 =(16D)
53 OCT 1 # ND1 =(17D)
```

|          |       |           |                                           |        |
|----------|-------|-----------|-------------------------------------------|--------|
|          | OCT   | 0         | # VD2                                     | =(18D) |
|          | OCT   | 0         | # VD1                                     | =(19D) |
| VERB     | CAF   | ZERO      |                                           |        |
|          | TS    | VERBREG   |                                           |        |
| NVCOM    | CAF   | VD1       |                                           |        |
|          | TS    | DSPCOUNT  |                                           |        |
|          | TC    | 2BLANK    |                                           |        |
|          | CAF   | ONE       |                                           |        |
|          | TS    | DECBRNCH  | # SET FOR DEC V/N CODE                    |        |
|          | CAF   | ZERO      |                                           |        |
|          | TS    | REQRET    | # SET FOR ENTPASO                         |        |
|          | CAF   | ENDINST   | # IF DSPALARM OCCURS BEFORE FIRST ENTPASO |        |
|          | TS    | ENTRET    | # OR NVSUB, ENTRET MUST ALREADY BE SET    |        |
|          |       |           | # TO TC ENDOFJOB                          |        |
| NOUN     | TC    | ENDOFJOB  |                                           |        |
|          | CAF   | ZERO      |                                           |        |
|          | TS    | NOUNREG   |                                           |        |
|          | CAF   | ND1       | # ND1, OCT 21 (DEC 17)                    |        |
|          | TC    | NVCOM     |                                           |        |
| NEGSGN   | TC    | SIGNTEST  |                                           |        |
|          | TC    | -ON       |                                           |        |
| BOTHSGN  | CAF   | TWO       |                                           |        |
|          | INDEX | INREL     | # SET DEC COMP BIT TO 1 (IN DECBRNCH)     |        |
|          | AD    | BIT7      | # BIT 5 FOR R1. BIT 4 FOR R2.             |        |
| FIXCLPAS | ADS   | DECBRNCH  | # BIT 3 FOR R3.                           |        |
|          | CCS   | CLPASS    | # IF CLPASS IS + OR +0. MAKE IT +0.       |        |
|          | CAF   | ZERO      |                                           |        |
|          | TS    | CLPASS    |                                           |        |
|          | TC    | +1        |                                           |        |
|          | TC    | ENDOFJOB  |                                           |        |
| POSGN    | TC    | SIGNTEST  |                                           |        |
|          | TC    | +ON       |                                           |        |
|          | CAF   | ONE       |                                           |        |
|          | TC    | BOTHSGN   |                                           |        |
| +ON      | LXCH  | Q         |                                           |        |
|          | TC    | GETINREL  |                                           |        |
|          | INDEX | INREL     |                                           |        |
|          | CAF   | SGNTAB -2 |                                           |        |
|          | TS    | SGNOFF    |                                           |        |
|          | AD    | ONE       |                                           |        |
| SGNCOM   | TS    | SGNON     |                                           |        |
|          | CAF   | ZERO      |                                           |        |
|          | TS    | CODE      |                                           |        |
|          | XCH   | SGNOFF    |                                           |        |

```
1
2 TC 11DSPIN
3 CAF BIT11
4 TS CODE
5 XCH SGNON
6 TC 11DSPIN
7 TC L
8 -ON LXCH Q
9 TC GETINREL
10
11 INDEX INREL
12 CAF SGNTAB -2
13 TS SGNON
14
15 AD ONE
16 TS SGNOFF
17 TC SGNCOM
18
19 SGNTAB OCT 5 # -R1
20 OCT 3 # -R2
21 OCT 0 # -R3
22
23 SIGNTEST LXCH Q # ALLOWS +,- ONLY WHEN DSPCOUNT=R1D1,
24 CAF THREE # R2D1, OR R3D1. ALLOWS ONLY FIRST OF
25 MASK DECBRNCH # CONSECUTIVE +/- CHARACTERS.
26 CCS A # IF LOW2 BITS OF DECBRNCH NOT= 0, SIGN
27 TC ENDOFJOB # FOR THIS WORD ALREADY IN. REJECT.
28 CS R1D1
29 TC SGNTST1
30 CS R2D1
31 TC SGNTST1
32 CS R3D1
33 SGNTST1 TC SGNTST1
34 TC ENDOFJOB # NO MATCH FOUND. SIGN ILLEGAL
35 AD DSPCOUNT
36
37 EXTEND
38 BZF +2 # MATCH FOUND
39 TC Q
40 TC L # SIGN LEGAL
41
42 # CLEAR BLANKS WHICH R1, R2, R3 IS CURRENT OR LAST TO BE DISPLAYED (PERTINENT
43 # XREG, YREG, ZREG IS CLEARED). SUCCESSIVE CLEARS TAKE CARE OF EACH RX
44 # L/ RC UNTIL R1 IS DONE. THEN NO FURTHER ACTION.
45 #
46 # THE SINGLE COMPONENT LOAD VERBS ALLOW ONLY THE SINGLE RC THAT IS
47 # APPROPRIATE TO BE CLEARED.
48 #
49 # CLPASS +0 PASS0, CAN BE BACKED UP
50 # +NZ HIPASS, CAN BE BACKED UP
51 # -NZ PASS0, CANNOT BE BACKED UP
```

```
1 CLEAR CCS DSPCOUNT
2 AD ONE
3 TC +2
4 AD ONE
5 INDEX A # DO NOT CHANGE DSPCOUNT BECAUSE MAY LATER
6 # FAIL LEGALTST.
7 CAF INRELTAB # MUST SET INREL, EVEN FOR HIPASS.
8 TS INREL
9 CCS CLPASS
10 TC CLPASHI # +
11 TC +2 # +0 IF CLPASS IS +0 OR -, IT IS PASS0
12 TC +1 # -
13 CA INREL
14 TC LEGALTST
15 TC CLEAR1
16 CLPASHI CCS INREL
17 TS INREL
18 TC LEGALTST
19 CAF DOUBLK +2 # +3 TO - NUMBER, BACKS DATA REQUESTS.
20 ADS REQRET
21 CA INREL
22 TS MIXTEMP # TEMP STORAGE FOR INREL
23 EXTEND
24 DIM VERBREG # DECREMENT VERB AND RE-DISPLAY
25 TC BANKCALL
26 CADR UPDATVB
27 CA MIXTEMP
28 CLEAR1 TS INREL # RESTORE INREL
29 TC CLR5
30 INCR CLPASS # ONLY IF CLPASS IS + OR +0.
31 CLR5 TC ENDOFJOB # SET FOR HIGHER PASS.
32 LXCH Q # USES 5BLANK BUT AVOIDS ITS TC GETINREL
33 TC 5BLANK +2
34 LEGALTST AD NEG2
35 CCS A
36 TC Q # LEGAL INREL G/ 2
37 TC CCSHOLE
38 TC ENDOFJOB # ILLEGAL INREL=0,1
39 TC Q # LEGAL INREL=2
40
41 # 5BLANK BLANKS 5 CHAR DISPLAY WORD IN R1, R2, OR R3. IT ALSO ZEROES XREG,
42 # YREG, OR ZREG. PLACE ANY + DSPCOUNT NUMBER FOR PERTINENT RC INTO DSPCOUNT.
43 # DSPCOUNT IS LEFT SET TO LEFT MOST DSP NUMB FOR RC JUST BLANKED.
44
45 TS DSPCOUNT # NEEDED FOR BLANKSUB
46 5BLANK LXCH Q
47 TC GETINREL
48 CAF ZERO
49 INDEX INREL
50 TS VERBREG # ZERO X, Y, Z, REG.
```

|    |   |
|----|---|
| 76 | 1 |
| 77 |   |

```
1 # ENTER PASS 0 IS THE EXECUTE FUNCTION. HIGHER ORDER ENTERS ARE TO LOAD
2 # DATA. THE SIGN OF REQRET DETERMINES THE PASS, + FOR PASS 0, - FOR HIGHER
3 # PASSES.
4 #
5 # MACHINE CADR TO BE SPECIFIED (MCTBS) NOUNS DESIRE AN ECADR TO BE LOADED
6 # WHEN USED WITH LOAD VERBS, MONITOR VERBS, OR DISPLAY VERBS (EXCEPT
7 # VERB = FIXED MEMORY DISPLAY, WHICH REQUIRES AN FCADR).
```

```
10 BANK 41
11 SETLOC PINBALL2
12 BANK
```

```
13
14 NVSUBB COUNT* $$/PIN
15 TC NVSUB1 # STANDARD LEAD INS. DONT MOVE.
16 LOADLV1 TC LOADLV
```

```
17
18 # END OF STANDARD LEAD INS.
```

```
19 ENTER CAF ZERO
20 TS CLPASS
```

```
21 CAF ENDINST
22 TS ENTRET
23 CCS REQRET
```

```
24 TC ENTPASO # IF +, PASS 0
25 TC ENTPASO # IF +, PASS 0
26 TC +1 # IF -, NOT PASS 0
```

```
27 ENTPASHI CAF MMADREF
28 AD REQRET # IF L/ 2 CHAR IN FOR MM CODE, ALARM
29 EXTEND # AND RECYCLE (DECIDE AT MMCHANG+1).
```

```
30 BZF ACCEPTWD
31 CAF THREE # IF DEC, ALARM IF L/ 5 CHAR IN FOR DATA,
32 MASK DECBRNCH # BUT LEAVE REQRET - AND FLASH ON, SO
33 CCS A # OPERATOR CAN SUPPLY MISSING NUMERICAL
34 TC +2 # CHARACTERS AND CONTINUE.
35 TC ACCEPTWD # OCTAL. ANY NUMBER OF CHAR OK.
```

```
36 CCS DSPCOUNT
37 TC GODSPALM # LESS THAN 5 CHAR DEC(DSPCOUNT IS +)
38 TC GODSPALM # LESS THAN 5 CHAR DEC(DSPCOUNT IS +)
```

```
39 ACCEPSTD TC +1 # 5 CHAR IN (DSPCOUNT IS -)
40 CS REQRET # 5 CHAR IN (DSPCOUNT IS -)
41 TS REQRET # SET REQRET +.
```

```
42 TC FLASHOFF
43 TC REQRET
```

```
44 ENTEXIT = ENTRET
```

```
45 MMADREF ADRES MMCHANG +1 # ASSUMES TC REQMM AT MMCHANG.
```



|         |        |             |                                           |
|---------|--------|-------------|-------------------------------------------|
| LOWVERB | DEC    | 28          | # LOWER VERB THAT AVOIDS NOUN TEST.       |
| ENTPASO | CAF    | ZERO        | # NOUN VERB SUB ENTERS HERE               |
|         | TS     | DECBRNCH    |                                           |
|         | CS     | VD1         | # BLOCK FURTHER NUM CHAR, SO THAT STRAY   |
| TESTVB  | TS     | DSPCOUNT    | # CHAR DO NOT GET INTO VERB OR NOUN LTS.  |
|         | CS     | VERBREG     | # IF VERB IS G/E LOWVB, SKIP NOUN TEST.   |
|         | TS     | VERBSAVE    | # SAVE VERB FOR POSSIBLE RECYCLE.         |
|         | AD     | LOWVERB     | # LOWVERB - VB                            |
|         | EXTEND |             |                                           |
|         | BZMF   | VERBFAN     | # VERB G/ E LOWVERB                       |
| TESTNN  | EXTEND |             | # VERB L/ LOWVERB                         |
|         | DCA    | LODNNLOC    | # SWITCH BANKS TO NOUN TABLE READING      |
|         | DXCH   | Z           | # ROUTINE.                                |
|         | INDEX  | MIXBR       |                                           |
|         | TC     | +0          |                                           |
|         | TC     | +2          | # NORMAL                                  |
|         | TC     | MIXNOUN     | # MIXED                                   |
|         | CCS    | NNADTEM     | # NORMAL                                  |
|         | TC     | VERBFAN -2  | # NORMAL IF +                             |
|         | TC     | GODSPALM    | # NOT IN USE IF +0                        |
|         | TC     | REQADD      | # SPECIFY MACHINE CADR IF -               |
|         | INCR   | NOUNCADR    | # AUGMENT MACHINE CADR IF -0              |
|         | TC     | SETNADD     | # ECADR FROM NOUNCADR, SETS EB, NOUNADD.  |
|         | TC     | INTMCTBS +2 |                                           |
| REQADD  | CAF    | BIT15       | # SET CLPASS FOR PASS 0 ONLY              |
|         | TS     | CLPASS      |                                           |
|         | CS     | ENDINST     | # TEST IF REACHED HERE FROM INTERNAL OR   |
|         | AD     | ENTEXIT     | # FROM EXTERNAL                           |
|         | EXTEND |             |                                           |
|         | BZF    | +2          | # EXTERNAL MACH CADR TO BE SPECIFIED      |
|         | TC     | INTMCTBS    |                                           |
|         | TC     | REQDATZ     | # EXTERNAL MACH CADR TO BE SPECIFIED      |
|         | CCS    | DECBRNCH    | # ALARM AND RECYCLE IF DECIMAL USED       |
|         | TC     | ALMCYCLE    | # FOR MCTBS.                              |
|         | CS     | VD1         | # OCTAL USED OK                           |
|         | TS     | DSPCOUNT    | # BLOCK NUM CHAR IN                       |
|         | CCS    | CADRSTOR    |                                           |
|         | TC     | +3          | # EXTERNAL MCTBS DISPLAY WILL LEAVE FLASH |
|         | TC     | USEADD      | # ON IF ENDIDLE NOT = +0.                 |
|         | TC     | +1          |                                           |
| USEADD  | TC     | FLASHON     |                                           |
|         | XCH    | ZREG        |                                           |
|         | TC     | SETNCADR    | # ECADR INTO NOUNCADR. SET EB, NOUNADD.   |
|         | EXTEND |             |                                           |
|         | DCA    | LODNNLOC    | # SWITCH BANKS TO NOUN TABLE READING      |
|         | DXCH   | Z           | # ROUTINE.                                |
|         | TC     | VERBFAN     |                                           |
|         | EBANK= | DSPCOUNT    |                                           |

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| LODNNLOC | 2CADR  | LODNNTAB |                                           |
| NEG5     | OCT    | 77772    |                                           |
| INTMCTBS | CA     | MPAC +2  | # INTERNAL MACH CADR TO BE SPECIFIED.     |
|          | TC     | SETNCADR | # ECADR INTO NOUNCADR. SET EB. NOUNADD.   |
|          | CS     | FIVE     | # NVSUB CALL LEFT CADR IN MPAC+2 FOR MACH |
|          | AD     | VERBREG  | # CADR TO BE SPECIFIED.                   |
|          | EXTEND |          |                                           |
|          | BZF    | VERBFAN  | # DONT DISPLAY CADR IF VB = 05.           |
|          | CAF    | R3D1     | # VB NOT = 05. DISPLAY CADR.              |
|          | TS     | DSPCOUNT |                                           |
|          | CA     | NOUNCADR |                                           |
|          | TC     | DSPOCTWO |                                           |
|          | TC     | VERBFAN  |                                           |
|          | AD     | ONE      |                                           |
| VERBFAN  | TC     | SETNCADR | # ECADR INTO NOUNCADR. SETS EB, NOUNADD.  |
|          | CS     | LST2CON  |                                           |
|          | AD     | VERBREG  | # VERB = LST2CON                          |
|          | CCS    | A        |                                           |
|          | AD     | ONE      | # VERB G/ LST2CON                         |
|          | TC     | +2       |                                           |
|          | TC     | VBFANDIR | # VERB L/ LST2CON                         |
|          | TS     | MPAC     |                                           |
|          | TC     | RELDSP   | # RELEASE DISPLAY SYST                    |
|          | TC     | POSTJUMP | # GO TO GOEXTVB WITH VB=40 IN MPAC.       |
| LST2CON  | CADR   | GOEXTVB  |                                           |
|          | DEC    | 40       | # FIRST LIST2 VERB (EXTENDED VERB)        |
| VBFANDIR | INDEX  | VERBREG  |                                           |
|          | CAF    | VERBTAB  |                                           |
|          | TC     | BANKJUMP |                                           |
| VERBTAB  | CADR   | GODSPALM | # VB00 ILLEGAL                            |
|          | CADR   | DSPA     | # VB01 DISPLAY OCT COMP 1 (R1)            |
|          | CADR   | DSPB     | # VB02 DISPLAY OCT COMP 2 (R1)            |
|          | CADR   | DSPC     | # VB03 DISPLAY OCT COMP 3 (R1)            |
|          | CADR   | DSPAB    | # VB04 DISPLAY OCT COMP 1,2 (R1,R2)       |
|          | CADR   | DSPABC   | # VB05 DISPLAY OCT COMP 1,2,3 (R1,R2,R3)  |
|          | CADR   | DECdsp   | # VB06 DECIMAL DISPLAY                    |
|          | CADR   | DSPDPDEC | # VB07 DP DECIMAL DISPLAY (R1,R2)         |
|          | CADR   | GODSPALM | # VB08 SPARE                              |
|          | CADR   | GODSPALM | # VB09 SPARE                              |
|          | CADR   | DSPALARM | # VB10 SPARE                              |
|          | CADR   | MONITOR  | # VB11 MONITOR OCT COMP 1 (R1)            |
|          | CADR   | MONITOR  | # VB12 MONITOR OCT COMP 2 (R1)            |
|          | CADR   | MONITOR  | # VB13 MONITOR OCT COMP 3 (R1)            |
|          | CADR   | MONITOR  | # VB14 MONITOR OCT COMP 1,2 (R1,R2)       |

|          |      |          |                                           |
|----------|------|----------|-------------------------------------------|
|          | CADR | MONITOR  | # VB15 MONITOR OCT COMP 1,2,3 (R1,R2,R3)  |
|          | CADR | MONITOR  | # VB16 MONITOR DECIMAL                    |
|          | CADR | MONITOR  | # VB17 MONITOR DP DEC (R1,R2)             |
|          | CADR | GODSPALM | # VB18 SPARE                              |
|          | CADR | GODSPALM | # VB19 SPARE                              |
|          | CADR | GODSPALM | # VB20 SPARE                              |
|          | CADR | ALOAD    | # VB21 LOAD COMP 1 (R1)                   |
|          | CADR | BLOAD    | # VB22 LOAD COMP 2 (R2)                   |
|          | CADR | CLOAD    | # VB23 LOAD COMP 3 (R3)                   |
|          | CADR | ABLOAD   | # VB24 LOAD COMP 1,2 (R1,R2)              |
|          | CADR | ABCLOAD  | # VB25 LOAD COMP 1,2,3 (R1,R2,R3)         |
|          | CADR | GODSPALM | # VB26 SPARE                              |
|          | CADR | DSPFMEM  | # VB27 FIXED MEMORY DISPLAY               |
|          |      |          | # THE FOLLOWING VERBS MAKE NO NOUN TEST   |
|          | CADR | GODSPALM | # VB28 SPARE                              |
|          | CADR | GODSPALM | # VB29 SPARE                              |
| REQEXLOC | CADR | VBRQEXEC | # VB30 REQUEST EXECUTIVE                  |
|          | CADR | VBRQWAIT | # VB31 REQUEST WAITLIST                   |
|          | CADR | VBRESEQ  | # VB32 RESEQUENCE                         |
|          | CADR | VBPROC   | # VB33 PROCEED WITHOUT DATA               |
|          | CADR | VBTERM   | # VB34 TERMINATE CURRENT TEST OR LOAD REQ |
|          | CADR | VBTSTLTS | # VB35 TEST LIGHTS                        |
|          | CADR | SLAP1    | # VB36 FRESH START                        |
|          | CADR | MMCHANG  | # VB37 CHANGE MAJOR MODE                  |
|          | CADR | GODSPALM | # VB38 SPARE                              |
|          | CADR | GODSPALM | # VB39 SPARE                              |

# THE LIST2 VERBFAN IS LOCATED IN THE EXTENDED VERB BANK.

# NNADTAB CONTAINS A RELATIVE ADDRESS, IDADDREL (IN LOW 10 BITS), REFERRING  
# TO WHERE 3 CONSECUTIVE ADDRESSES ARE STORED (IN IDADDTAB).

# MIXNOUN GETS DATA AND STORES IN MIXTEMP,+1,+2. IT SETS NOUNADD FOR  
# MIXTEMP.

|         |     |          |                 |
|---------|-----|----------|-----------------|
| MIXNOUN | CCS | NNADTEM  |                 |
|         | TC  | +4       | # + IN USE      |
|         | TC  | GODSPALM | # +0 NOT IN USE |
|         | TC  | +2       | # - IN USE      |
|         | TC  | +1       | # -0 IN USE     |
|         | CS  | SIX      |                 |

|  |        |         |              |
|--|--------|---------|--------------|
|  | AD     | VERBREG |              |
|  | EXTEND |         |              |
|  | BZMF   | +2      | # VERB L/E 6 |

|  |    |         |                                          |
|--|----|---------|------------------------------------------|
|  | TC | VERBFAN | # AVOID MIXNOUN SWAP IF VB NOT = DISPLAY |
|--|----|---------|------------------------------------------|

|        |     |         |  |
|--------|-----|---------|--|
| MIXNN1 | CAF | TWO     |  |
|        | TS  | DECOUNT |  |

|  |       |          |                                      |
|--|-------|----------|--------------------------------------|
|  | AD    | MIXAD    |                                      |
|  | TS    | NOUNADD  | # SET NOUNADD TO MIXTEMP + K         |
|  | INDEX | DECOUNT  | # GET IDADDTAB ENTRY FOR COMPONENT K |
|  | CA    | IDADITEM | # OF NOUN.                           |
|  | TS    | NOUNTEM  |                                      |

|  |    |          |                                             |
|--|----|----------|---------------------------------------------|
|  |    |          | # TEST FOR DP (FOR OCT DISPLAY). IF SO, GET |
|  |    |          | # MINOR PART ONLY.                          |
|  | TC | SFRUTMIX | # GET SF ROUT NUMBER IN A                   |
|  | TC | DPTEST   |                                             |

|        |      |         |                     |
|--------|------|---------|---------------------|
|        | TC   | MIXNN2  | # NO DP             |
|        | INCR | NOUNTEM | # DP GET MINOR PART |
| MIXNN2 | CA   | NOUNTEM |                     |

|  |       |          |                                               |
|--|-------|----------|-----------------------------------------------|
|  | MASK  | LOW11    | # ESUBK (NO DP) OR (ESUBK)+1 (GARBLED) FOR DP |
|  | TC    | SETEBANK | # SET EBANK, LEAVE EADRES IN A.               |
|  | INDEX | A        | # PICK UP C(ESUBK) NOT DP                     |
|  | CA    | 0        | # OR C((ESUBK)+1) FOR DP MINOR PART           |

|  |       |         |                       |
|--|-------|---------|-----------------------|
|  | INDEX | NOUNADD |                       |
|  | XCH   | 0       | # STORE IN MIXTEM + K |

|  |     |         |  |
|--|-----|---------|--|
|  | CCS | DECOUNT |  |
|  | TC  | MIXNN1  |  |
|  | TC  | VERBFAN |  |

|       |    |         |  |
|-------|----|---------|--|
| MIXAD | TC | MIXTEMP |  |
|-------|----|---------|--|

# DPTTEST ENTER WITH SF ROUT NUMBER IN A.  
# RETURNS TO L+1 IF NO DP.  
# RETURNS TO L+2 IF DP.

|         |       |    |  |
|---------|-------|----|--|
| DPTTEST | INDEX | A  |  |
|         | TCF   | +1 |  |

|  |    |   |                    |
|--|----|---|--------------------|
|  | TC | Q | # OCTAL ONLY NO DP |
|  | TC | Q | # FRACT NO DP      |

|          |        |           |                                          |
|----------|--------|-----------|------------------------------------------|
|          | TC     | Q         | # DEG NO DP                              |
|          | TC     | Q         | # ARITH NO DP                            |
|          | TCF    | DPTEST1   | # DP1OUT                                 |
|          | TCF    | DPTEST1   | # DP2OUT                                 |
|          | TC     | Q         | # LRPOSOUT NO DP (DATA IN CHANNEL 33)    |
|          | TCF    | DPTEST1   | # DP3OUT                                 |
|          | TC     | Q         | # HMS NO DP                              |
|          | TC     | Q         | # M/S NO DP                              |
|          | TCF    | DPTEST1   | # DP4OUT                                 |
|          | TC     | Q         | # ARITH1 NO DP                           |
|          | TC     | Q         | # 2INTOUT NO DP TO GET HI PART IN MPAC   |
|          | TC     | Q         | # 360-CDU NO DP                          |
| DPTTEST1 | INDEX  | Q         |                                          |
|          | TC     | 1         | # RETURN TO L+2                          |
| REQDATX  | CAF    | R1D1      |                                          |
|          | TCF    | REQCOM    |                                          |
| REQDATY  | CAF    | R2D1      |                                          |
|          | TCF    | REQCOM    |                                          |
| REQDATZ  | CAF    | R3D1      |                                          |
| REQCOM   | TS     | DSPCOUNT  |                                          |
|          | CS     | Q         |                                          |
|          | TS     | REQRET    |                                          |
|          | TC     | BANKCALL  |                                          |
|          | CADR   | 5BLANK    |                                          |
|          | TC     | FLASHON   |                                          |
| ENDRQDAT | TC     | ENTEXIT   |                                          |
|          | TS     | NOUNREG   |                                          |
| UPDATNN  | XCH    | Q         |                                          |
|          | TS     | UPDATRET  |                                          |
|          | EXTEND |           |                                          |
|          | DCA    | LODNNLOC  | # SWITCH BANKS TO NOUN TABLE READING     |
|          | DXCH   | Z         | # ROUTINE.                               |
|          | CCS    | NNADTEM   |                                          |
|          | AD     | ONE       | # NORMAL                                 |
|          | TCF    | PUTADD    |                                          |
|          | TCF    | PUTADD +1 | # MCTBS DONT CHANGE NOUNADD              |
|          | TCF    | PUTADD +1 | # MCTBI DONT CHANGE NOUNADD              |
| PUTADD   | TC     | SETNCADR  | # ECADR INTO NOUNCADR. SETS EB. NOUNADD. |
|          | CAF    | ND1       |                                          |
|          | TS     | DSPCOUNT  |                                          |
|          | CA     | NOUNREG   |                                          |
|          | TCF    | UPDAT1    |                                          |
|          | TS     | VERBREG   |                                          |
| UPDATVB  | XCH    | Q         |                                          |
|          | TS     | UPDATRET  |                                          |
|          | CAF    | VD1       |                                          |

|    |                                      |      |          |                                            |    |
|----|--------------------------------------|------|----------|--------------------------------------------|----|
| 1  | # FINDERALL CANAL BUTTONS AND LIGHTS |      |          |                                            | 1  |
| 2  |                                      | TS   | DSPCOUNT |                                            | 2  |
| 3  |                                      | CA   | VERBREG  |                                            | 3  |
| 4  | UPDAT1                               | TC   | POSTJUMP | # CANT USE SWCALL TO GO TO DSPDECVN, SINCE | 4  |
| 5  |                                      | CADR | GOVNUPDT | # UPDATVB CAN ITSELF BE CALLED BY SWCALL.  | 5  |
| 6  |                                      | TC   | UPDATRET |                                            | 6  |
| 7  |                                      |      |          |                                            | 7  |
| 8  | GOALMCYC                             | TC   | ALMCYCLE | # NEEDED BECAUSE BANKJUMP CANT HANDLE F/F. | 8  |
| 9  |                                      |      |          |                                            | 9  |
| 10 | GODSPALM                             | TC   | POSTJUMP |                                            | 10 |
| 11 |                                      | CADR | DSPALARM |                                            | 11 |
| 12 |                                      |      |          |                                            | 12 |
| 13 |                                      |      |          |                                            | 13 |
| 14 |                                      |      |          |                                            | 14 |
| 15 |                                      |      |          |                                            | 15 |
| 16 |                                      |      |          |                                            | 16 |
| 17 |                                      |      |          |                                            | 17 |
| 18 |                                      |      |          |                                            | 18 |
| 19 |                                      |      |          |                                            | 19 |
| 20 |                                      |      |          |                                            | 20 |
| 21 |                                      |      |          |                                            | 21 |
| 22 |                                      |      |          |                                            | 22 |
| 23 |                                      |      |          |                                            | 23 |
| 24 |                                      |      |          |                                            | 24 |
| 25 |                                      |      |          |                                            | 25 |
| 26 |                                      |      |          |                                            | 26 |
| 27 |                                      |      |          |                                            | 27 |
| 28 |                                      |      |          |                                            | 28 |
| 29 |                                      |      |          |                                            | 29 |
| 30 |                                      |      |          |                                            | 30 |
| 31 |                                      |      |          |                                            | 31 |
| 32 |                                      |      |          |                                            | 32 |
| 33 |                                      |      |          |                                            | 33 |
| 34 |                                      |      |          |                                            | 34 |
| 35 |                                      |      |          |                                            | 35 |
| 36 |                                      |      |          |                                            | 36 |
| 37 |                                      |      |          |                                            | 37 |
| 38 |                                      |      |          |                                            | 38 |
| 39 |                                      |      |          |                                            | 39 |
| 40 |                                      |      |          |                                            | 40 |
| 41 |                                      |      |          |                                            | 41 |
| 42 |                                      |      |          |                                            | 42 |
| 43 |                                      |      |          |                                            | 43 |
| 44 |                                      |      |          |                                            | 44 |
| 45 |                                      |      |          |                                            | 45 |
| 46 |                                      |      |          |                                            | 46 |
| 47 |                                      |      |          |                                            | 47 |
| 48 |                                      |      |          |                                            | 48 |
| 49 |                                      |      |          |                                            | 49 |
| 50 |                                      |      |          |                                            | 50 |
| 51 |                                      |      |          |                                            | 51 |
| 52 |                                      |      |          |                                            | 52 |
| 53 |                                      |      |          |                                            | 53 |
| 54 |                                      |      |          |                                            | 54 |
| 55 |                                      |      |          |                                            | 55 |
| 56 |                                      |      |          |                                            | 56 |
| 57 |                                      |      |          |                                            | 57 |
| 58 |                                      |      |          |                                            | 58 |
| 59 |                                      |      |          |                                            | 59 |
| 60 |                                      |      |          |                                            | 60 |

## # NOUN TABLES

#

# NOUN CODE L/40, NORMAL NOUN CASE. NOUN CODE G/E 40, MIXED NOUN CASE.

# FOR NORMAL CASE, NNADTAB CONTAINS ONE ECADR FOR EACH NOUN.

# +0 INDICATES NOUN NOT USED. - ENTRY INDICATES MACHINE CADR (E OR F) TO

# BE SPECIFIED. -1 INDICATES CHANNEL TO BE SPECIFIED. -0 INDICATES AUGMENT

# OF LAST MACHINE CADR SUPPLIED.

#

# FOR MIXED CASE, NNADTAB CONTAINS ONE INDIRECT ADDRESS (IDADDREL) IN LOW

# 10 BITS, AND THE COMPONENT CODE NUMBER IN THE HIGH 5 BITS.

#

# NNTYPTAB IS A PACKED TABLE OF THE FORM MMMMMNNNNNPPPPP.

#

# FOR THE NORMAL CASE, M'S ARE THE COMPONENT CODE NUMBER.

# N'S ARE THE SF ROUTINE CODE NUMBER.

# P'S ARE THE SF CONSTANT CODE NUMBER.

#

# MIXED-CASE, M'S ARE THE SF CONSTANT3 CODE NUMBER 3 COMPONENT CASE

# N'S ARE THE SF CONSTANT2 CODE NUMBER

# P'S ARE THE SF CONSTANT1 CODE NUMBER

# N'S ARE THE SF CONSTANT2 CODE NUMBER 2 COMPONENT CASE

# P'S ARE THE SF CONSTANT1 CODE NUMBER

# P'S ARE THE SF CONSTANT1 CODE NUMBER 1 COMPONENT CASE

#

# THERE IS ALSO AN INDIRECT ADDRESS TABLE (IDADDTAB) FOR MIXED CASE ONLY.

# EACH ENTRY CONTAINS ONE ECADR. IDADDREL IS THE RELATIVE ADDRESS OF

# THE FIRST OF THESE ENTRIES.

#

# THERE IS ALSO A SCALE FACTOR ROUTINE NUMBER TABLE (RUTMXTAB) FOR MIXED

# CASE ONLY. THERE IS ONE ENTRY PER MIXED NOUN. THE FORM IS,

#

# QQQQRRRRRSSSSS

#

# Q'S ARE THE SF ROUTINE 3 CODE NUMBER 3 COMPONENT CASE

# R'S ARE THE SF ROUTINE 2 CODE NUMBER

# S'S ARE THE SF ROUTINE 1 CODE NUMBER

# R'S ARE THE SF ROUTINE 2 CODE NUMBER 2 COMPONENT CASE

# S'S ARE THE SF ROUTINE 1 CODE NUMBER

#

# IN OCTAL DISPLAY AND LOAD (OCT OR DEC) VERBS, EXCLUDE USE OF VERBS WHOSE

# COMPONENT NUMBER IS GREATER THAN THE NUMBER OF COMPONENTS IN NOUN.

# (ALL MACHINE ADDRESS TO BE SPECIFIED NOUNS ARE 3 COMPONENT.)

#

# IN MULTI-COMPONENT LOAD VERBS, NO MIXING OF OCTAL AND DECIMAL DATA

# COMPONENT WORDS IS ALLOWED. ALARM IF VIOLATION.

#

# IN DECIMAL LOADS OF DATA, 5 NUMERICAL CHARACTERS MUST BE KEYED IN

# BEFORE EACH ENTER. IF NOT, ALARM.

# DISPLAY VERBS

|         |       |            |                                 |  |  |  |  |  |  |
|---------|-------|------------|---------------------------------|--|--|--|--|--|--|
| DSPABC  | CS    | TWO        |                                 |  |  |  |  |  |  |
|         | TC    | COMPTST    |                                 |  |  |  |  |  |  |
|         | INDEX | NOUNADD    |                                 |  |  |  |  |  |  |
|         | CS    | 2          |                                 |  |  |  |  |  |  |
| DSPAB   | XCH   | BUF        | +2                              |  |  |  |  |  |  |
|         | CS    | ONE        |                                 |  |  |  |  |  |  |
|         | TC    | COMPTST    |                                 |  |  |  |  |  |  |
|         | INDEX | NOUNADD    |                                 |  |  |  |  |  |  |
|         | CS    | 1          |                                 |  |  |  |  |  |  |
| DSPA    | XCH   | BUF        | +1                              |  |  |  |  |  |  |
|         | TC    | DECTEST    |                                 |  |  |  |  |  |  |
|         | TC    | TSTFORDP   |                                 |  |  |  |  |  |  |
|         | INDEX | NOUNADD    |                                 |  |  |  |  |  |  |
| DSPCOM1 | CS    | 0          |                                 |  |  |  |  |  |  |
|         | XCH   | BUF        |                                 |  |  |  |  |  |  |
| DSPB    | TC    | DSPCOM2    |                                 |  |  |  |  |  |  |
|         | CS    | ONE        |                                 |  |  |  |  |  |  |
|         | TC    | DCOMPTST   |                                 |  |  |  |  |  |  |
|         | INDEX | NOUNADD    |                                 |  |  |  |  |  |  |
|         | CS    | 1          |                                 |  |  |  |  |  |  |
|         | TC    | DSPCOM1    |                                 |  |  |  |  |  |  |
| DSPC    | CS    | TWO        |                                 |  |  |  |  |  |  |
|         | TC    | DCOMPTST   |                                 |  |  |  |  |  |  |
|         | INDEX | NOUNADD    |                                 |  |  |  |  |  |  |
|         | CS    | 2          |                                 |  |  |  |  |  |  |
|         | TC    | DSPCOM1    |                                 |  |  |  |  |  |  |
| DSPCOM2 | CS    | TWO        | # A B C AB ABC                  |  |  |  |  |  |  |
|         | AD    | VERBREG    | # -1 -0 +1 +2 +3 IN A           |  |  |  |  |  |  |
|         | CCS   | A          | # +0 +0 +0 +1 +2 IN A AFTER CCS |  |  |  |  |  |  |
|         | TC    | DSPCOM3    |                                 |  |  |  |  |  |  |
|         | TC    | ENTEXIT    |                                 |  |  |  |  |  |  |
|         | TC    | +1         |                                 |  |  |  |  |  |  |
| DSPCOM3 | TS    | DISTEM     | # +0 +1 +2 INTO DISTEM          |  |  |  |  |  |  |
|         | INDEX | A          |                                 |  |  |  |  |  |  |
|         | CAF   | R1D1       |                                 |  |  |  |  |  |  |
|         | TS    | DSPCOUNT   |                                 |  |  |  |  |  |  |
|         | INDEX | DISTEM     |                                 |  |  |  |  |  |  |
|         | CS    | BUF        |                                 |  |  |  |  |  |  |
|         | TC    | DSPOCTWO   |                                 |  |  |  |  |  |  |
|         | XCH   | DISTEM     |                                 |  |  |  |  |  |  |
|         | TC    | DSPCOM2 +2 |                                 |  |  |  |  |  |  |

# COMPTST ALARMS IF COMPONENT NUMBER OF VERB (LOAD OR OCT DISPLAY) IS  
# GREATER THAN THE HIGHEST COMPONENT NUMBER OF NOUN.

|          |      |         |             |
|----------|------|---------|-------------|
| COMPTST  | TS   | SFTEMP1 | # VERB COMP |
|          | LXCH | Q       |             |
| COMPTST1 | TC   | GETCOMP |             |
|          | TC   | LEFT5   |             |
|          | MASK | THREE   | # NOUN COMP |



```
1
2 AD SFTEMP1 # NOUN COMP = VERB COMP
3 CCS A
4 TC L # NOUN COMP G/ VERB COMP
5 TC CCSHOLE
6 TC GODSPALM # NOUN COMP L/ VERB COMP
7 NDCMPTST TC L # NOUN COMP = VERB COMP
8
9 # DCOMPTST ALARMS IF DECIMAL ONLY BIT (BIT4 OF COMP CODE NUMBER) = 1.
10 # IF NOT, IT PERFORMS REGULAR COMPTST.
11
12 DCOMPTST TS SFTEMP1 # - VERB COMP
13 LXCH Q
14 TC DECTEST
15 TC COMPTST1
16
17 DECTEST EXTEND
18 QXCH MPAC +2 # ALARMS IF DEC ONLY BIT = 1 (BIT4 OF COMP
19 # CODE NUMBER). RETURNS IF NOT.
20 TC GETCOMP
21 MASK BIT14
22 CCS A
23 TC GODSPALM
24 TC MPAC +2
25
26 DCTSTCYC LXCH Q # ALARMS AND RECYCLES IF DEC ONLY BIT = 1
27 TC GETCOMP # (BIT4 OF COMP CODE NUMBER). RETURNS
28 MASK BIT14 # IF NOT. USED BY LOAD VERBS.
29 CCS A
30 TC ALMCYCLE
31 TC L
32
33 # NOUNTEST ALARMS IF NO-LOAD BIT (BIT5 OF COMP CODE NUMBER) = 1.
34 # IF NOT, IT RETURNS.
35
36 NOUNTEST LXCH Q
37 TC GETCOMP
38 CCS A
39 TC L
40 TC L
41 TC GODSPALM
42
43 TSTFORDP LXCH Q # TEST FOR DP. IF SO, GET MINOR PART ONLY.
44 CA NNADTEM
45 AD ONE # IF NNADTEM = -1, CHANNEL TO BE SPECIFIED
46 EXTEND
47 BZF CHANDSP
48 INDEX MIXBR
49 TC +0
50 TC +2 # NORMAL
51
52
53
54
55
56
57
58
59
60
```

|    |          |        |            |                                         |                  |  |    |
|----|----------|--------|------------|-----------------------------------------|------------------|--|----|
| 1  |          |        |            |                                         |                  |  | 1  |
| 2  |          | TC     | L          | # MIXED CASE ALREADY HANDLED IN MIXNOUN |                  |  | 2  |
| 3  |          | TC     | SFRUTNOR   |                                         |                  |  | 3  |
| 4  |          | TC     | DPTEST     |                                         |                  |  | 4  |
| 5  |          | TC     | L          | # NO DP                                 |                  |  | 5  |
| 6  |          | INCR   | NOUNADD    | # DP E+1 INTO NOUNADD FOR MINOR PART.   |                  |  | 6  |
| 7  |          | TC     | L          |                                         |                  |  | 7  |
| 8  |          |        |            |                                         |                  |  | 8  |
| 9  | CHANDSP  | CA     | NOUNCADR   |                                         |                  |  | 9  |
| 10 |          | MASK   | LOW9       |                                         |                  |  | 10 |
| 11 |          | EXTEND |            |                                         |                  |  | 11 |
| 12 |          | INDEX  | A          |                                         |                  |  | 12 |
| 13 |          | READ   | 0          |                                         |                  |  | 13 |
| 14 |          | CS     | A          |                                         |                  |  | 14 |
| 15 |          | TCF    | DSPCOM1    |                                         |                  |  | 15 |
| 16 |          |        |            |                                         |                  |  | 16 |
| 17 | COMPICK  | ADRES  | NNTYPTM    |                                         |                  |  | 17 |
| 18 |          | ADRES  | NNADTEM    |                                         |                  |  | 18 |
| 19 |          |        |            |                                         |                  |  | 19 |
| 20 | GETCOMP  | INDEX  | MIXBR      | # NORMAL                                | MIXED            |  | 20 |
| 21 |          | CAF    | COMPICK -1 | # ADRES NNTYPTM                         | ADRES NNADTEM    |  | 21 |
| 22 |          | INDEX  | A          |                                         |                  |  | 22 |
| 23 |          | CA     | 0          | # C(NNTYPTM)                            | C(NNADTEM)       |  | 23 |
| 24 |          | MASK   | HI5        | # GET HI5 OF NNTYPTAB (NORM)            | OF NNADTAB (MIX) |  | 24 |
| 25 |          | TC     | Q          |                                         |                  |  | 25 |
| 26 |          |        |            |                                         |                  |  | 26 |
| 27 | DECDSP   | TC     | GETCOMP    |                                         |                  |  | 27 |
| 28 |          | TC     | LEFT5      |                                         |                  |  | 28 |
| 29 |          | MASK   | THREE      |                                         |                  |  | 29 |
| 30 |          | TS     | DECOUNT    | # COMP NUMBER INTO DECOUNT              |                  |  | 30 |
| 31 | DSPDCGET | TS     | DECTEM     | # PICKS UP DATA                         |                  |  | 31 |
| 32 |          | AD     | NOUNADD    | # DECTEM 1COMP +0, 2COMP +1, 3COMP +2   |                  |  | 32 |
| 33 |          | INDEX  | A          |                                         |                  |  | 33 |
| 34 |          | CS     | 0          |                                         |                  |  | 34 |
| 35 |          | INDEX  | DECTEM     |                                         |                  |  | 35 |
| 36 |          | XCH    | XREG       | # CANT USE BUF SINCE DMP USES IT.       |                  |  | 36 |
| 37 |          | CCS    | DECTEM     |                                         |                  |  | 37 |
| 38 |          | TC     | DSPDCGET   | # MORE TO GET                           |                  |  | 38 |
| 39 | DSPDCPUT | CAF    | ZERO       | # DISPLAYS DATA                         |                  |  | 39 |
| 40 |          | TS     | MPAC +1    | # DECOUNT 1COMP +0, 2COMP +1, 3COMP +2  |                  |  | 40 |
| 41 |          | TS     | MPAC +2    |                                         |                  |  | 41 |
| 42 |          | INDEX  | DECOUNT    |                                         |                  |  | 42 |
| 43 |          | CAF    | R1D1       |                                         |                  |  | 43 |
| 44 |          | TS     | DSPCOUNT   |                                         |                  |  | 44 |
| 45 |          | INDEX  | DECOUNT    |                                         |                  |  | 45 |
| 46 |          | CS     | XREG       |                                         |                  |  | 46 |
| 47 |          | TS     | MPAC       |                                         |                  |  | 47 |
| 48 |          | TC     | SFCONUM    | # 2X (SF CON NUMB) IN A                 |                  |  | 48 |
| 49 |          |        |            |                                         |                  |  | 49 |
| 50 |          |        |            |                                         |                  |  | 50 |
| 51 |          |        |            |                                         |                  |  | 51 |
| 52 |          |        |            |                                         |                  |  | 52 |
| 53 |          |        |            |                                         |                  |  | 53 |
| 54 |          |        |            |                                         |                  |  | 54 |
| 55 |          |        |            |                                         |                  |  | 55 |
| 56 |          |        |            |                                         |                  |  | 56 |
| 57 |          |        |            |                                         |                  |  | 57 |
| 58 |          |        |            |                                         |                  |  | 58 |
| 59 |          |        |            |                                         |                  |  | 59 |
| 60 |          |        |            |                                         |                  |  | 60 |

|    |                                                                       |        |          |                                          |    |
|----|-----------------------------------------------------------------------|--------|----------|------------------------------------------|----|
| 1  |                                                                       |        |          |                                          | 1  |
| 2  |                                                                       | TS     | SFTEMP1  |                                          | 2  |
| 3  |                                                                       | EXTEND |          | # SWITCH BANKS TO SF CONSTANT TABLE      | 3  |
| 4  |                                                                       | DCA    | GTSFOUTL | # READING ROUTINE.                       | 4  |
| 5  |                                                                       | DXCH   | Z        | # LOADS SFTEMP1, SFTEMP2                 | 5  |
| 6  |                                                                       | INDEX  | MIXBR    |                                          | 6  |
| 7  |                                                                       | TC     | +0       |                                          | 7  |
| 8  |                                                                       | TC     | DSPSFNOR |                                          | 8  |
| 9  |                                                                       | TC     | SFRUTMIX |                                          | 9  |
| 10 |                                                                       | TC     | DECDSP3  |                                          | 10 |
| 11 |                                                                       |        |          |                                          | 11 |
| 12 | DSPSFNOR                                                              | TC     | SFRUTNOR |                                          | 12 |
| 13 |                                                                       | TC     | DECDSP3  |                                          | 13 |
| 14 |                                                                       |        |          |                                          | 14 |
| 15 |                                                                       | EBANK= | DSPCOUNT |                                          | 15 |
| 16 | GTSFOUTL                                                              | 2CADR  | GTSFOUT  |                                          | 16 |
| 17 |                                                                       |        |          |                                          | 17 |
| 18 | DSPDCEND                                                              | TC     | BANKCALL | # ALL SFOUT ROUTINES END HERE            | 18 |
| 19 |                                                                       | CADR   | DSPDECWD |                                          | 19 |
| 20 |                                                                       | CCS    | DECOUNT  |                                          | 20 |
| 21 |                                                                       | TC     | +2       |                                          | 21 |
| 22 |                                                                       | TC     | ENTEXIT  |                                          | 22 |
| 23 |                                                                       | TS     | DECOUNT  |                                          | 23 |
| 24 |                                                                       | TC     | DSPDCPUT | # MORE TO DISPLAY                        | 24 |
| 25 |                                                                       |        |          |                                          | 25 |
| 26 | DECDSP3                                                               | INDEX  | A        |                                          | 26 |
| 27 |                                                                       | CAF    | SFOUTABR |                                          | 27 |
| 28 |                                                                       | TC     | BANKJUMP |                                          | 28 |
| 29 |                                                                       |        |          |                                          | 29 |
| 30 | SFOUTABR                                                              | CADR   | PREDSPAL | # ALARM IF DEC DISP WITH OCTAL ONLY NOUN | 30 |
| 31 |                                                                       | CADR   | DSPDCEND |                                          | 31 |
| 32 |                                                                       | CADR   | DEGOUTSF |                                          | 32 |
| 33 |                                                                       | CADR   | ARTOUTSF |                                          | 33 |
| 34 |                                                                       | CADR   | DP1OUTSF |                                          | 34 |
| 35 |                                                                       | CADR   | DP2OUTSF |                                          | 35 |
| 36 |                                                                       | CADR   | LRPOSOUT |                                          | 36 |
| 37 |                                                                       | CADR   | DP3OUTSF |                                          | 37 |
| 38 |                                                                       | CADR   | HMSOUT   |                                          | 38 |
| 39 |                                                                       | CADR   | M/SOUT   |                                          | 39 |
| 40 |                                                                       | CADR   | DP2OUTSF |                                          | 40 |
| 41 |                                                                       | CADR   | AROUT1SF |                                          | 41 |
| 42 |                                                                       | CADR   | 2INTOUT  |                                          | 42 |
| 43 |                                                                       | CADR   | 360-CDUO |                                          | 43 |
| 44 | ENDRTOUT                                                              | EQUALS |          |                                          | 44 |
| 45 |                                                                       |        |          |                                          | 45 |
| 46 | # THE FOLLOWING IS A TYPICAL SF ROUTINE. IT USES MPAC. LEAVES RESULTS |        |          |                                          | 46 |
| 47 | # IN MPAC, MPAC+1. ENDS WITH TC DSPDCEND                              |        |          |                                          | 47 |
| 48 |                                                                       |        |          |                                          | 48 |
| 49 |                                                                       |        |          |                                          | 49 |
| 50 |                                                                       |        |          |                                          | 50 |
| 51 |                                                                       |        |          |                                          | 51 |
| 52 |                                                                       |        |          |                                          | 52 |
| 53 |                                                                       |        |          |                                          | 53 |
| 54 |                                                                       |        |          |                                          | 54 |
| 55 |                                                                       |        |          |                                          | 55 |
| 56 |                                                                       |        |          |                                          | 56 |
| 57 |                                                                       |        |          |                                          | 57 |
| 58 |                                                                       |        |          |                                          | 58 |
| 59 |                                                                       |        |          |                                          | 59 |
| 60 |                                                                       |        |          |                                          | 60 |

SETLOC BLANKCON +1

COUNT\* \$\$/PIN

# DEGOUTSF SCALES BY .18 THE LOW 14 BITS OF ANGLE, ADDING .18 FOR  
# NUMBERS IN THE NEGATIVE (AGC) RANGE.

DEGOUTSF CAF ZERO  
TS MPAC +2 # SET INDEX FOR FULL SCALE.

TC FIXRANGE

TC +2

# NO AUGMENT NEEDED (SFTEMP1 AND 2 ARE 0)

TC SETAUG

# SET AUGMENTER ACCORDING TO C(MPAC +2)

TC DEGCOM

# 360-CDUD COMPUTES 360 - CDU ANGLE IN MPAC, STORES RESULT IN MPAC AND  
# GOES TO DEGOUTSF.

360-CDUD TC 360-CDU  
TC DEGOUTSF

360-CDU CA MPAC

MASK POSMAX

# IF ANGLE IS 0 OR 180 DEGREES, DO NOTHING

EXTEND

BZF 360-CDUE

CS MPAC

# COMPUTE 360 DEGREES MINUS ANGLE

AD ONE

TS MPAC

360-CDUE TC Q

# LRPOSOUT DISPLAYS +0,1,2, OR 3 (WHOLE) FOR CHANNEL 33, BITS 7-6 = 11,10,  
# 01,00 RESPECTIVELY

LRPOSOUT EXTEND  
READ CHAN33

EXTEND

MP BIT10

# BITS 7-6 TO BITS 2-1

COM

MASK THREE

TS MPAC

TC ARTOUTSF # DISPLAY AS WHOLE

SETAUG EXTEND # LOADS SFTEMP1 AND SFTEMP2 WITH THE  
# DP AUGMENTER CONSTANT

INDEX MPAC +2

DCA DEGTAB

DXCH SFTEMP1

TC Q

FIXRANGE CCS MPAC # IF MPAC IS + RETURN TO L+1

TC Q

# IF MPAC IS - RETURN TO L+2 AFTER

TC Q

# MASKING OUT THE SIGN BIT

TCF +1

|    |          |        |          |                                            |  |
|----|----------|--------|----------|--------------------------------------------|--|
| 1  |          |        |          |                                            |  |
| 2  |          | CS     | BIT15    |                                            |  |
| 3  |          | MASK   | MPAC     |                                            |  |
| 4  |          | TS     | MPAC     |                                            |  |
| 5  |          | INDEX  | Q        |                                            |  |
| 6  |          | TC     | 1        |                                            |  |
| 7  |          |        |          |                                            |  |
| 8  | DEGCOM   | EXTEND |          | # LOADS MULTIPLIER, DOES SHORTMP, AND      |  |
| 9  |          | INDEX  | MPAC +2  | # ADDS AUGMENTER.                          |  |
| 10 |          | DCA    | DEGTAB   |                                            |  |
| 11 |          | DXCH   | MPAC     | # ADJUSTED ANGLE IN A                      |  |
| 12 |          | TC     | SHORTMP  |                                            |  |
| 13 |          | DXCH   | SFTEMP1  |                                            |  |
| 14 |          | DAS    | MPAC     |                                            |  |
| 15 |          | TC     | SCOUTEND |                                            |  |
| 16 |          |        |          |                                            |  |
| 17 | DEGTAB   | OCT    | 05605    | # HI PART OF .18                           |  |
| 18 |          | OCT    | 03656    | # LOW PART OF .18                          |  |
| 19 |          | OCT    | 16314    | # HI PART OF .45                           |  |
| 20 |          | OCT    | 31463    | # LO PART OF .45                           |  |
| 21 |          |        |          |                                            |  |
| 22 | ARTOUTSF | DXCH   | SFTEMP1  | # ASSUMES POINT AT LEFT OF DP SFCON        |  |
| 23 |          | DXCH   | MPAC     |                                            |  |
| 24 |          | TC     | PRSHRTMP | # IF C(A) = -0, SHORTMP FAILS TO GIVE -0.  |  |
| 25 | SCOUTEND | TC     | POSTJUMP |                                            |  |
| 26 |          | CADR   | DSPDCEND |                                            |  |
| 27 |          |        |          |                                            |  |
| 28 | AROUT1SF | DXCH   | SFTEMP1  | # ASSUMES POINT BETWEEN HI AND LO PARTS OF |  |
| 29 |          | DXCH   | MPAC     | # DP SFCON. SHIFTS RESULTS LEFT 14, BY     |  |
| 30 |          | TC     | PRSHRTMP | # TAKING RESULTS FROM MPAC+1, MPAC+2.      |  |
| 31 |          | TC     | L14/OUT  |                                            |  |
| 32 |          |        |          |                                            |  |
| 33 | DP1OUTSF | TC     | DPOUT    | # SCALES MPAC, MPAC +1 BY DP SCALE FACTOR  |  |
| 34 | L14/OUT  | XCH    | MPAC +2  | # IN SFTEMP1, SFTEMP2. THEN SCALE RESULT   |  |
| 35 |          | XCH    | MPAC +1  | # BY B14                                   |  |
| 36 |          | TS     | MPAC     |                                            |  |
| 37 |          | TC     | SCOUTEND |                                            |  |
| 38 |          |        |          |                                            |  |
| 39 | DP2OUTSF | TC     | DPOUT    | # SCALES MPAC, MPAC +1 BY DP SCALE FACTOR  |  |
| 40 |          | TC     | SCOUTEND |                                            |  |
| 41 |          |        |          |                                            |  |
| 42 | DP3OUTSF | TC     | DPOUT    | # ASSUMES POINT BETWEEN BITS 7-8 OF HIGH   |  |
| 43 |          | CAF    | SIX      | # LEFT BY 7, ROUNDS MPAC+2 INTO MPAC+1.    |  |
| 44 |          | TC     | TPLEFTN  | # SHIFT LEFT 7.                            |  |
| 45 |          | TC     | SCOUTEND |                                            |  |
| 46 |          |        |          |                                            |  |
| 47 |          |        |          |                                            |  |
| 48 |          |        |          |                                            |  |
| 49 |          |        |          |                                            |  |
| 50 |          |        |          |                                            |  |
| 51 |          |        |          |                                            |  |
| 52 |          |        |          |                                            |  |
| 53 |          |        |          |                                            |  |
| 54 |          |        |          |                                            |  |
| 55 |          |        |          |                                            |  |
| 56 |          |        |          |                                            |  |
| 57 |          |        |          |                                            |  |
| 58 |          |        |          |                                            |  |
| 59 |          |        |          |                                            |  |
| 60 |          |        |          |                                            |  |

```
1 MPAC+6 = MPAC +6 # USE MPAC +6 INSTEAD OF OVFIN
2
3
4 DPOUT XCH Q
5 TS MPAC+6
6 TC READLO # GET FRESH DATA FOR BOTH HI AND LO.
7 TC TPAGREE # MAKE DP DATA AGREE
8 TC DMP
9 ADRES SFTEMP1
10 TC MPAC+6
11
12 # THE FOLLOWING ROUTINE DISPLAYS TWO CONTIGUOUS SP POSITIVE INTEGERS
13 # AS TWO POSITIVE DECIMAL INTEGERS IN RXD1-RXD2 AND RXD4-RXD5 (RXD3 IS
14 # BLANKED). THE INTEGER IN THE LOWER NUMBERED ADDRESS IS DISPLAYED IN
15 # RXD1-RXD2.
16
17 2INTOUT TC 5BLANK # TO BLANK RXD3
18 TC +ON # TURN ON + SIGN
19
20 CA MPAC
21 TC DSPDECVN # DISPLAY 1ST INTEGER (LIKE VERB AND NOUN)
22 CS THREE
23 INDEX DECOUNT
24 AD R1D1 # RXD4
25 TS DSPCOUNT
26
27 TC READLO # GET 2ND INTEGER
28 CA MPAC +1
29 TC DSPDECVN # DISPLAY 2ND INTEGER (LIKE VERB AND NOUN)
30 TC POSTJUMP
31 CADR DSPDCEND +2
32
33 # READLO PICKS UP FRESHDATA FOR BOTH HI AND LO AND LEAVES IT IN
34 # MPAC, MPAC+1. THIS IS NEEDED FOR TIME DISPLAY. IT ZEROES MPAC+2, BUT
35 # DOES NOT FORCE TPAGREE.
36
37 READLO XCH Q
38 TS TEM4
39 INDEX MIXBR
40 TC +0
41 TC RDLONOR
42 INDEX DECOUNT
43 CA IDAD1TEM # GET IDADDTAB ENTRY FOR COMP K OF NOUN.
44 MASK LOW11 # E SUBK
45 TC SETEBANK # SET EB, LEAVE EADRES IN A.
46 EXTEND # MIXED NORMAL
47 INDEX A # C(ESUBK) C(E)
48 DCA 0 # C(E SUBK)+1) C(E+1)
49 DXCH MPAC
50 CAF ZERO
51 TS MPAC +2
52 TC TEM4
53
54
55
56
57
58
59
60
```

|         |        |                      |                                             |
|---------|--------|----------------------|---------------------------------------------|
| RDONOR  | CA     | NOUNADD              | # E                                         |
| ENDRDLO | TC     | READLO1              |                                             |
|         | BANK   | 42                   |                                             |
|         | SETLOC | PINBALL3             |                                             |
|         | BANK   |                      |                                             |
|         | COUNT* | \$\$/PIN             |                                             |
| HMSOUT  | TC     | BANKCALL             | # READ FRESH DATA FOR HI AND LO INTO MPAC,  |
|         | CADR   | READLO               | # MPAC+1.                                   |
|         | TC     | TPAGREE              | # MAKE DP DATA AGREE.                       |
|         | TC     | SEPSECNR             | # LEAVE FRACT SEC/60 IN MPAC, MPAC+1. LEAVE |
|         |        |                      | # WHOLE MIN IN BIT13 OF LOTEMOUT AND ABOVE  |
|         | TC     | DMP                  | # USE ONLY FRACT SEC/60 MOD 60              |
|         | ADRES  | SECON2               | # MULT BY .06                               |
|         | CAF    | R3D1                 | # GIVES CENTI-SEC/10EXP5 MOD 60             |
|         | TS     | DSPCOUNT             |                                             |
|         | TC     | BANKCALL             | # DISPLAY SEC MOD 60                        |
|         | CADR   | DSPDECWD             |                                             |
|         | TC     | SEPMIN               | # REMOVE REST OF SECONDS                    |
|         | CAF    | MINCON2              | # LEAVE FRACT MIN/60 IN MPAC+1. LEAVE       |
|         | XCH    | MPAC                 | # WHOLE HOURS IN MPAC.                      |
|         | TS     | HITEMOUT             | # SAVE WHOLE HOURS.                         |
|         | CAF    | MINCON2 +1           |                                             |
|         | XCH    | MPAC +1              | # USE ONLY FRACT MIN/60 MOD 60              |
|         | TC     | PRSHRTMP             | # IF C(A) = -0, SHORTMP FAILS TO GIVE -0.   |
|         |        |                      | # MULT BY .0006                             |
|         | CAF    | R2D1                 | # GIVE MIN/10EXP5 MOD 60                    |
|         | TS     | DSPCOUNT             |                                             |
|         | TC     | BANKCALL             | # DISPLAY MIN MOD 60                        |
|         | CADR   | DSPDECWD             |                                             |
|         | EXTEND |                      | # MINUTES, SECONDS HAVE BEEN REMOVED        |
|         | DCA    | HRCON1               |                                             |
|         | DXCH   | MPAC                 |                                             |
|         | CA     | HITEMOUT             | # USE WHOLE HOURS                           |
|         | TC     | PRSHRTMP             | # IF C(A) = -0, SHORTMP FAILS TO GIVE -0.   |
|         |        |                      | # MULT BY .16384                            |
|         | CAF    | R1D1                 | # GIVES HOURS/10EXP5                        |
|         | TS     | DSPCOUNT             |                                             |
|         | TC     | BANKCALL             | # USE REGULAR DSPDECWD, WITH ROUND OFF.     |
|         | CADR   | DSPDECWD             |                                             |
|         | TC     | ENTEXIT              |                                             |
| SECON1  | 2DEC*  | 1.666666666 E-4 B12* | # 2EXP12/6000                               |
| SECON2  | OCT    | 01727                | # .06 FOR SECONDS DISPLAY                   |
|         | OCT    | 01217                |                                             |
| MINCON2 | OCT    | 00011                | # .0006 FOR MINUTES DISPLAY                 |
|         | OCT    | 32445                |                                             |

|          |      |             |                                            |
|----------|------|-------------|--------------------------------------------|
| MINCON1  | OCT  | 02104       | # .066..66 UPPED BY 2EXP-28                |
|          | OCT  | 10422       |                                            |
| HRCON1   | 2DEC | .16384      |                                            |
|          | OCT  | 00000       |                                            |
| RNDCON   | OCT  | 00062       | # .5 SEC                                   |
| M/SOUT   | TC   | BANKCALL    | # READ FRESH DATA FOR HI AND LO INTO MPAC. |
|          | CADR | READLO      | # MPAC+1.                                  |
|          | TC   | TPAGREE     | # MAKE DP DATA AGREE                       |
|          | CCS  | MPAC        | # IF MAG OF (MPAC, MPAC+1) G/ 59 M 59 S.   |
|          | TC   | +2          | # DISPLAY 59B59, WITH PROPER SIGN.         |
|          | TC   | M/SNORM     | # MPAC = +0. L/ 59M58.5S                   |
|          | AD   | M/SCON1     | # - HI PART OF (59M58.5S) +1 FOR CCS       |
|          | CCS  | A           | # MAG OF MPAC - HI PART OF (59M58.5S)      |
|          | TC   | M/SLIMIT    | # G/ 59M58.5S                              |
|          | TC   | M/SNORM     | # ORIGINAL MPAC = -0. L/ 59M58.5S          |
|          | TC   | M/SNORM     | # L/ 59M58.5S                              |
|          | CCS  | MPAC +1     | # MAG OF MPAC = HI PART OF 59M58.5S        |
|          | TC   | +2          |                                            |
|          | TC   | M/SNORM     | # MPAC+1 = +0. L/ 59M58.5S                 |
|          | AD   | M/SCON2     | # - LO PART OF (59M58.5S) +1 FOR CCS       |
|          | CCS  | A           | # MAG OF MPAC+1 - LO PART OF (59M58.5S)    |
|          | TC   | M/SLIMIT    | # G/ 59M58.5S                              |
|          | TC   | M/SNORM     | # ORIGINAL MPAC+1 = -0. L/ 59M58.5S        |
|          | TC   | M/SNORM     | # L/ 59M58.5S                              |
| M/SLIMIT | CCS  | MPAC        | # = 59M58.5S LIMIT                         |
|          | CAF  | M/SCON3     | # MPAC CANNOT BE +/- 0 AT THIS POINT.      |
|          | TC   | +LIMIT      | # FORCE MPAC, MPAC+1 TO +/- 59M59.5S       |
|          | CS   | M/SCON3     |                                            |
|          | TS   | MPAC        | # WILL DISPLAY 59M59S IN DSPDECNR          |
|          | CS   | M/SCON3 +1  |                                            |
| LIMITCOM | TS   | MPAC +1     |                                            |
|          | CAF  | NORMADR     | # SET RETURN TO M/SNORM+1.                 |
|          | TC   | SEPSECNR +1 |                                            |
| +LIMIT   | TS   | MPAC        |                                            |
|          | CAF  | M/SCON3 +1  |                                            |
|          | TC   | LIMITCOM    |                                            |
| M/SNORM  | TC   | SEPSEC      | # LEAVE FRACT SEC/60 IN MPAC,MPAC+1. LEAVE |
|          |      |             | # WHOLE MIN IN BIT13 OF LOTEMOUT AND ABOVE |
|          | CAF  | HISECON     | # USE ONLY FRACT SEC/60 MOD 60             |
|          | TC   | SHORTMP     | # MULT BY .6 + 2EXP-14                     |
|          | CS   | THREE       | # GIVES SEC/100 MOD 60                     |
|          | ADS  | DSPCOUNT    | # DSPCOUNT ALREADY SET TO RXD1             |
|          | TC   | BANKCALL    | # DISPLAY SEC MOD 60 IN D4D5.              |
|          | CADR | DSPDC2NR    |                                            |
|          | CAF  | ZERO        |                                            |
|          | TS   | CODE        |                                            |
|          | CS   | TWO         |                                            |



|          |        |             |                                         |
|----------|--------|-------------|-----------------------------------------|
|          | INDEX  | DECOUNT     |                                         |
|          | AD     | R1D1        | # RXD3                                  |
|          | TS     | COUNT       |                                         |
|          | TC     | BANKCALL    | # BLANK MIDDLE CHAR                     |
|          | CADR   | DSPIN       |                                         |
|          | TC     | SEPMIN      | # REMOVE REST OF SECONDS                |
|          | XCH    | MPAC +1     | # LEAVE FRACT MIN/60 IN MPAC+1          |
|          | EXTEND |             | # USE ONLY FRACT MIN/60 MOD 60          |
|          | MP     | HIMINCON    | # MULT BY .6 + 2EXP-7                   |
|          | DXCH   | MPAC        | # GIVES MIN/100 MOD 60                  |
|          | INDEX  | DECOUNT     |                                         |
|          | CAF    | R1D1        | # RXD1                                  |
|          | TS     | DSPCOUNT    |                                         |
|          | TC     | BANKCALL    | # DISPLAY MIN MOD 60 IN D1D2.           |
|          | CADR   | DSPDC2NR    |                                         |
|          | TC     | POSTJUMP    |                                         |
|          | CADR   | DSPDCEND +2 |                                         |
| HISECON  | OCT    | 23147       | # .6 + 2EXP-14                          |
| HIMINCON | OCT    | 23346       | # .6 + 2EXP-7                           |
| M/SCON1  | OCT    | 77753       | # - HI PART OF (59M58.5S) +1            |
| M/SCON2  | OCT    | 41126       | # - LO PART OF (59M58.5S) +1            |
| NORMADR  | ADRES  | M/SNORM +1  |                                         |
| M/SCON3  | OCT    | 00025       | # 59M 59.5S                             |
|          | OCT    | 37016       |                                         |
| SEPSEC   | CCS    | MPAC +1     | # IF +, ROUND BY ADDING .5 SEC          |
|          | TCF    | POSEC       | # IF -, ROUND BY SUBTRACING .5 SEC      |
|          | TCF    | POSEC       | # FINDS TIME IN MPAC, MPAC+1            |
|          | TCF    | +1          | # ROUNDS OFF BY +/- .5 SEC              |
|          | EXTEND |             | # LEAVES WHOLE MIN IN BIT13 OF          |
| SEPSEC1  | DCS    | RNDCON -1   | # LOTEMOUT AND ABOVE.                   |
|          | DAS    | MPAC        | # LEAVES FRACT SEC/60 IN MPAC, MPAC+1.  |
|          | TCF    | SEPSECNR    |                                         |
| POSEC    | EXTEND |             |                                         |
|          | DCA    | RNDCON -1   |                                         |
|          | TCF    | SEPSEC1     |                                         |
| SEPSECNR | XCH    | Q           | # THIS ENTRY AVOIDS ROUNDING BY .5 SEC  |
|          | TS     | SEPSCRET    |                                         |
|          | TC     | DMP         | # MULT BY 2EXP12/6000                   |
|          | ADRES  | SECON1      | # GIVES FRACT SEC/60 IN BIT12 OF MPAC+1 |
|          | EXTEND |             | # AND BELOW.                            |
|          | DCA    | MPAC        | # SAVE MINUTES AND HOURS                |
|          | DXCH   | HITEMOUT    |                                         |
|          | TC     | TPSL1       |                                         |
|          | TC     | TPSL1       | # GIVES FRACT SEC/60 IN MPAC+1, MPAC+2. |
|          | CAF    | ZERO        |                                         |
|          | XCH    | MPAC +2     | # LEAVE FRACT SEC/60 IN MPAC, MPAC+1.   |

```

1
2 XCH MPAC +1
3 XCH MPAC
4 TC SEPSECRET
5
6 SEPMIN XCH Q # FIND WHOLE MINUTES IN BIT13
7 TS SEPMNRET # OF LOTEMOUT AND ABOVE.
8 CA LOTEMOUT # REMOVES REST OF SECONDS.
9 EXTEND # LEAVES FRACT MIN/60 IN MPAC+1.
10 MP BIT3 # LEAVES WHOLE HOURS IN MPAC.
11 EXTEND # SR 12, THROW AWAY LP.
12 MP BIT13 # SR 2, TAKE FROM LP. = SL 12.
13 LXCH MPAC +1 # THIS FORCES BITS 12-1 TO 0 IF +.
14 # FORCES BITS 12-1 TO 1 IF -.
15 CA HITEMOUT
16 TS MPAC
17 TC DMP # MULT BY 1/15
18 ADRES MINCON1 # GIVES FRACT MIN/60 IN MPAC+1.
19 ENDSPMIN TC SEPMNRET # GIVES WHOLE HOURS IN MPAC.
20
21 # THIS IS A SPECIAL PURPOS VERB FOR DISPLAYING A DOUBLE PRECISION AGC
22 # WORD AS 10 DECIMAL DIGITS ON THE AGC DISPLAY PANEL. IT CAN BE USED WITH
23 # ANY NOUN, EXCEPT MIXED NOUNS. IT DISPLAYS THE CONTENTS
24 # OF THE REGISTER NOUNADD IS POINTING TO. IF USED WITH NOUNS WHICH ARE
25 # INHERENTLY NOT DP SUCH AS THE CDU COUNTERS THE DISPLAY WILL BE GARBAGE.
26 # DISPLAY IS IN R1 AND R2 ONLY WITH THE SIGN IN R1.
27
28 SETLOC ENDRDLO +1
29
30 COUNT* $$/PIN
31 DSPDPDEC INDEX MIXBR
32 TC +0
33 TC +2 # NORMAL NOUN
34 TC DSPALARM
35 EXTEND
36 INDEX NOUNADD
37 DCA 0
38 DXCH MPAC
39 CAF R1D1
40 TS DSPCOUNT
41 CAF ZERO
42 TS MPAC +2
43 TC TPAGREE
44 TC DSP2DEC
45 ENDDPDEC TC ENTEXIT

```

```
1 # LOAD VERBS IF ALARM CONDITION IS DETECTED DURING EXECUTE,
2 # CHECK FAIL LIGHT IS TURNED ON AND ENDOFJOB. IF ALARM CONDITION IS
3 # DETECTED DURING ENTER OF DATA, CHECK FAIL IS TURNED ON AND IT RECYCLES
4 # TO EXECUTE OF ORIGINAL LOAD VERB. RECYCLE CAUSED BY 1) DECIMAL MACHINE
5 # CADR 2) MIXTURE OF OCTAL/DECIMAL DATA 3) OCTAL DATA INTO DECIMAL
6 # ONLY NOUN 4) DEC DATA INTO OCT ONLY NOUN 5) DATA TOO LARGE FOR SCALE
7 # 6) FEWER THAN 3 DATA WORDS LOADED FOR HRS, MIN, SEC NOUN. (2)-(6) ALARM
8 # AND RECYCLE OCCUR AT FINAL ENTER OF SET. (1) ALARM AND RECYCLE OCCUR AT
9 # ENTER OF CADR.
```

```
11
12 SETLOC ENDRTOUT
```

```
13
14 COUNT* $$/PIN
15 ABCLOAD CS TWO
16 TC COMPTST
17 TC NOUNTEST # TEST IF NOUN CAN BE LOADED.
18 CAF VBSP1LD
19 TC UPDATVB -1
20 TC REQDATX
21 CAF VBSP2LD
22 TC UPDATVB -1
23 TC REQDATY
24 CAF VBSP3LD
25 TC UPDATVB -1
26 TC REQDATZ
27
28 PUTXYZ CS SIX # TEST THAT THE 3 DATA WORDS LOADED ARE
29 TC ALLDC/OC # ALL DEC OR ALL OCT.
30 EXTEND
31 DCA LODNNLOC # SWITCH BANKS TO NOUN TABLE READING
32 DXCH Z # ROUTINE.
33 CAF ZERO # X COMP
34 TC PUTCOM
35 INDEX NOUNADD
36 TS 0
37 CAF ONE # Y COMP
38 TC PUTCOM
39 INDEX NOUNADD
40 TS 1
41 CAF TWO # Z COMP
42 TC PUTCOM
43 INDEX NOUNADD
44 TS 2
45 CS SEVEN # IF NOUN 7 HAS JUST BEEN LOADED, SET
46 AD NOUNREG # FLAG BITS AS SPECIFIED.
47 EXTEND
48 BZF +2
49 TC LOADLV
```

|          |        |             |                                             |
|----------|--------|-------------|---------------------------------------------|
|          | CA     | XREG        | # ECADR OF FLAG WORD.                       |
|          | TC     | SETNCADR +1 | # SET EBANK, NOUNADD.                       |
|          | CA     | ZREG        | # ZERO TO RESET BITS, NON-ZERO TO SET BITS. |
|          | INHINT |             |                                             |
|          | EXTEND |             |                                             |
|          | BZF    | BITSOFF     |                                             |
|          | INDEX  | NOUNADD     |                                             |
|          | CS     | 0           |                                             |
|          | MASK   | YREG        | # BITS TO BE PROCESSED.                     |
|          | INDEX  | NOUNADD     |                                             |
|          | ADS    | 0           | # SET BITS.                                 |
| BITSOFF  | TC     | BITSOFF1    |                                             |
|          | CS     | YREG        | # BITS TO BE PROCESSED.                     |
|          | INDEX  | NOUNADD     |                                             |
|          | MASK   | 0           |                                             |
|          | INDEX  | NOUNADD     |                                             |
|          | TS     | 0           | # RESET BITS.                               |
| BITSOFF1 | RELINT |             |                                             |
|          | TC     | LOADLV      |                                             |
| ABLOAD   | CS     | ONE         |                                             |
|          | TC     | COMPTST     |                                             |
|          | TC     | NOUNTEST    | # TEST IF NOUN CAN BE LOADED.               |
|          | CAF    | VBSP1LD     |                                             |
|          | TC     | UPDATVB -1  |                                             |
|          | TC     | REQDATX     |                                             |
|          | CAF    | VBSP2LD     |                                             |
|          | TC     | UPDATVB -1  |                                             |
|          | TC     | REQDATY     |                                             |
| PUTXY    | CS     | FIVE        | # TEST THAT THE 2 DATA WORDS LOADED ARE     |
|          | TC     | ALLDC/OC    | # ALL DEC OR ALL OCT.                       |
|          | EXTEND |             |                                             |
|          | DCA    | LODNNLOC    | # SWITCH BANKS TO NOUN TABLE READING        |
|          | DXCH   | Z           | # ROUTINE.                                  |
|          | CAF    | ZERO        | # X COMP                                    |
|          | TC     | PUTCOM      |                                             |
|          | INDEX  | NOUNADD     |                                             |
|          | TS     | 0           |                                             |
|          | CAF    | ONE         | # Y COMP                                    |
|          | TC     | PUTCOM      |                                             |
|          | INDEX  | NOUNADD     |                                             |
|          | TS     | 1           |                                             |
|          | TC     | LOADLV      |                                             |
| ALOAD    | TC     | REQDATX     |                                             |
|          | EXTEND |             |                                             |
|          | DCA    | LODNNLOC    | # SWITCH BANKS TO NOUN TABLE READING        |
|          | DXCH   | Z           | # ROUTINE.                                  |
|          | CAF    | ZERO        | # X COMP                                    |
|          | TC     | PUTCOM      |                                             |

|    |          |        |          |                                           |
|----|----------|--------|----------|-------------------------------------------|
| 1  |          | INDEX  | NOUNADD  |                                           |
| 2  |          | TS     | 0        |                                           |
| 3  |          | TC     | LOADLV   |                                           |
| 4  |          |        |          |                                           |
| 5  |          |        |          |                                           |
| 6  | BLOAD    | CS     | ONE      |                                           |
| 7  |          | TC     | COMPTST  |                                           |
| 8  |          | CAF    | BIT15    | # SET CLPASS FOR PASSO ONLY               |
| 9  |          | TS     | CLPASS   |                                           |
| 10 |          | TC     | REQDATY  |                                           |
| 11 |          | EXTEND |          |                                           |
| 12 |          | DCA    | LODNNLOC | # SWITCH BANKS TO NOUN TABLE READING      |
| 13 |          | DXCH   | Z        | # ROUTINE.                                |
| 14 |          | CAF    | ONE      |                                           |
| 15 |          | TC     | PUTCOM   |                                           |
| 16 |          | INDEX  | NOUNADD  |                                           |
| 17 |          | TS     | 1        |                                           |
| 18 |          | TC     | LOADLV   |                                           |
| 19 |          |        |          |                                           |
| 20 | CLOAD    | CS     | TWO      |                                           |
| 21 |          | TC     | COMPTST  |                                           |
| 22 |          | CAF    | BIT15    | # SET CLPASS FOR PASSO ONLY               |
| 23 |          | TS     | CLPASS   |                                           |
| 24 |          | TC     | REQDATZ  |                                           |
| 25 |          | EXTEND |          |                                           |
| 26 |          | DCA    | LODNNLOC | # SWITCH BANKS TO NOUN TABLE READING      |
| 27 |          | DXCH   | Z        | # ROUTINE.                                |
| 28 |          | CAF    | TWO      |                                           |
| 29 |          | TC     | PUTCOM   |                                           |
| 30 |          | INDEX  | NOUNADD  |                                           |
| 31 |          | TS     | 2        |                                           |
| 32 |          | TC     | LOADLV   |                                           |
| 33 |          |        |          |                                           |
| 34 | LOADLV   | CAF    | ZERO     |                                           |
| 35 |          | TS     | DECBRNCH |                                           |
| 36 |          | CS     | ZERO     |                                           |
| 37 |          | TS     | LOADSTAT |                                           |
| 38 |          | TC     | RELDSP   | # RELEASE FOR PRIORITY DISPLAY PROBLEM.   |
| 39 |          | CS     | VD1      | # TO BLOCK NUMERICAL CHARACTERS AND       |
| 40 |          | TS     | DSPCOUNT | # CLEARS AFTER A COMPLETED LOAD           |
| 41 |          | TC     | POSTJUMP | # AFTER COMPLETED LOAD, GO TO RECALTST    |
| 42 |          | CADR   | RECALTST | # TO SEE IF THERE IS RECALL FROM ENDIDLE. |
| 43 |          |        |          |                                           |
| 44 | VBSP1LD  | DEC    | 21       | # VB21 = ALOAD                            |
| 45 | VBSP2LD  | DEC    | 22       | # VB22 = BLOAD                            |
| 46 | VBSP3LD  | DEC    | 23       | # VB23 = CLOAD                            |
| 47 |          |        |          |                                           |
| 48 | ALLDC/OC | TS     | DECOUNT  | # TESTS THAT DATA WORDS LOADED ARE EITHER |
| 49 |          | CS     | DECBRNCH | # ALL DEC OR ALL OCT. ALARMS IF NOT.      |
| 50 |          | TS     | SR       |                                           |
| 51 |          |        |          |                                           |
| 52 |          |        |          |                                           |
| 53 |          |        |          |                                           |
| 54 |          |        |          |                                           |
| 55 |          |        |          |                                           |
| 56 |          |        |          |                                           |
| 57 |          |        |          |                                           |
| 58 |          |        |          |                                           |
| 59 |          |        |          |                                           |
| 60 |          |        |          |                                           |

```
1
2 CS SR
3 CS SR # SHIFTED RIGHT 2
4 CCS A # DEC COMP BITS IN LOW 3
5 TCF +2 # SOME ONES IN LOW 3
6 TC Q # ALL ZEROS. ALL OCTAL. OK
7 AD DECOUNT # DEC COMP = 7 FOR 3COMP, =6 FOR 2COMP
8 EXTEND
9 BZF +2 # (BUT IT HAS BEEN DECREMENTED BY CCS)
10 TC ALMCYCLE # MUST MATCH 6 FOR 3COMP, 5 FOR 2COMP.
11 GOQ TC Q # ALARM AND RECYCLE.
12
13 SFRUTNOR XCH Q # GETS SF ROUTINE NUMBER FOR NORMAL CASE
14 TS EXITEM # CAN'T USE L FOR RETURN. TSTFORDP USES L.
15 CAF MID5
16
17 MASK NNTYPTM
18 TC RIGHT5
19 TC EXITEM # SF ROUTINE NUMBER IN A
20
21 SFRUTMIX XCH Q # GETS SF ROUTINE NUMBER FOR MIXED CASE
22 TS EXITEM
23 INDEX DECOUNT
24 CAF DISPLACE # PUT TC GOQ, TC RIGHT5, OR TC LEFT5 IN L
25 TS L
26
27 INDEX DECOUNT
28 CAF LOW5 # LOW5, MID5, OR HI5 IN A
29 MASK RUTMXTEM # GET HI5, MID5, OR LOW5 OF RUTMTAB ENTRY
30 INDEX L
31 TC 0
32
33 # DO TC GOQ(DECOUNT=0), DO TC RIGHT5(DECOUNT=1), DO TC LEFT5(DECOUNT=2).
34 SFRET1 TC EXITEM # SF ROUTINE NUMBER IN A
35
36 SFCONUM XCH Q # GETS 2X(SF CONSTANT NUMBER)
37 TS EXITEM
38 INDEX MIXBR
39
40 TC +0
41 TC CONUMNOR # NORMAL NOUN
42 INDEX DECOUNT # MIXED NOUN
43
44 CAF DISPLACE
45 TS L # PUT TC GOQ, TC RIGHT5, OR TC LEFT5 IN L
46 INDEX DECOUNT
47
48 CAF LOW5
49 MASK NNTYPTM
50 INDEX L
51 TC 0
52
53 # DO TC GOQ(DECOUNT=0), DO TC RIGHT5(DECOUNT=1), DO TC LEFT5(DECOUNT=2).
54 SFRET DOUBLE # 2X(SF CONSTANT NUMBER) IN A
55 TC EXITEM
56
57 DISPLACE TC GOQ
```

|                                                                           |        |          |                                          |
|---------------------------------------------------------------------------|--------|----------|------------------------------------------|
|                                                                           | TC     | RIGHT5   |                                          |
|                                                                           | TC     | LEFT5    |                                          |
| CONUMNOR                                                                  | CAF    | LOW5     | # NORMAL NOUN ALWAYS GETS LOW5 OF        |
|                                                                           | MASK   | NNTYPTM  | # NNTYPTAB FOR SF CONUM.                 |
|                                                                           | DOUBLE |          |                                          |
|                                                                           | TC     | EXITEM   | # 2X(SF CONSTANT NUMBER) IN A            |
| PUTCOM                                                                    | TS     | DECOUNT  |                                          |
|                                                                           | XCH    | Q        |                                          |
|                                                                           | TS     | DECRET   |                                          |
|                                                                           | CAF    | ZERO     |                                          |
|                                                                           | TS     | MPAC+6   |                                          |
|                                                                           | INDEX  | DECOUNT  |                                          |
|                                                                           | XCH    | XREGLP   |                                          |
|                                                                           | TS     | MPAC +1  |                                          |
|                                                                           | INDEX  | DECOUNT  |                                          |
|                                                                           | XCH    | XREG     |                                          |
|                                                                           | TS     | MPAC     |                                          |
|                                                                           | INDEX  | MIXBR    |                                          |
|                                                                           | TC     | +0       |                                          |
|                                                                           | TC     | PUTNORM  | # NORMAL NOUN                            |
| # IF MIXNOUN, PLACE ADDRESS FOR COMPONENT K INTO NOUNADD, SET EBANK BITS. |        |          |                                          |
|                                                                           | INDEX  | DECOUNT  | # GET IDADDTAB ENTRY FOR COMPONENT K     |
|                                                                           | CA     | IDADITEM | # OF NOUN.                               |
|                                                                           | MASK   | LOW11    | # (ECADR)SUBK FOR CURRENT COMP OF NOUN   |
|                                                                           | TC     | SETNCADR | # ECADR INTO NOUNCADR. SETS EB, NOUNADD. |
|                                                                           | EXTEND |          | # C(NOUNADD) IN A UPON RETURN            |
|                                                                           | SU     | DECOUNT  | # PLACE (ESUBK)-K INTO NOUNADD           |
|                                                                           | TS     | NOUNADD  |                                          |
|                                                                           | CCS    | DECBRNCH |                                          |
|                                                                           | TC     | PUTDECSF | # + DEC                                  |
|                                                                           | TC     | DCTSTCYC | # +0 OCTAL                               |
|                                                                           | TC     | SFRUTMIX | # TEST IF DEC ONLY BIT = 1. IF SO,       |
|                                                                           | TC     | DPTEST   | # ALARM AND RECYCLE. IF NOT, CONTINUE.   |
|                                                                           | TC     | PUTCOM2  | # NO DP                                  |
|                                                                           |        |          | # TEST FOR DP SCALE FOR OCT LOAD. IF SO, |
|                                                                           |        |          | # +0 INTO MAJOR PART. SET NOUNADD FOR    |
|                                                                           |        |          | # LOADING OCTAL WORD INTO MINOR PART.    |
| PUTDPCOM                                                                  | INCR   | NOUNADD  | # DP (ESUBK)-K+1 OR E+1                  |
|                                                                           | CA     | NOUNADD  | # NOUNADD NOW SET FOR MINOR PART         |
|                                                                           | ADS    | DECOUNT  | # (ESUBK)+1 OR E+1 INTO DECOUNT          |
|                                                                           | CAF    | ZERO     | # NOUNADD SET FOR MINOR PART             |
|                                                                           | INDEX  | DECOUNT  |                                          |
|                                                                           | TS     | 0 -1     | # ZERO MAJOR PART(ESUBK OR E)            |
|                                                                           | TC     | PUTCOM2  |                                          |
| PUTNORM                                                                   | TC     | SETNADD  | # ECADR FROM NOUNCADR. SETS EB, NOUNADD. |
|                                                                           | CCS    | DECBRNCH |                                          |

```
1
2 TC PUTDECSF # + DEC
3 TC DCTSTCYC # +0 OCTAL
4 TC SFRUTNOR # TEST IF DEC ONLY BIT =1. IF SO,
5 TC DPTEST # ALARM AND RECYCLE. IF NOT, CONTINUE.
6 TC PUTCOM2 -4 # NO DP
7 CAF ZERO # DP
8 TS DECOUNT
9 TC PUTDPCOM
10
11 CA NNADTEM
12 AD ONE # IF NNADTEM = -1, CHANNEL TO BE SPECIFIED
13
14 PUTCOM2 EXTEND
15 BZF CHANLOAD
16 XCH MPAC
17 TC DECRET
18
19 EBANK= DSPCOUNT
20 2CADR GTSFIN
21
22 CHANLOAD CS SEVEN # DONT LOAD CHAN 7. (IT = SUPERBANK).
23 AD NOUNCADR
24 EXTEND
25 BZF LOADLV
26 CA NOUNCADR
27 MASK LOW9
28 XCH MPAC
29 EXTEND
30 INDEX MPAC
31 WRITE 0
32 TC LOADLV
33
34 # PUTDECSF FINDS MIXBR AND DECOUNT STILL SET FROM PUTCOM
35
36 PUTDECSF TC SFCONUM # 2X(SF CON NUMB) IN A
37 TS SFTEMP1
38 EXTEND
39 DCA GTSFINLC # SWITCH BANKS TO SF CONSTANT TABLE
40 DXCH Z # READING ROUTINE.
41 INDEX MIXBR # LOADS SFTEMP1, SFTEMP2.
42 TC +0
43 TC PUTSFNOR
44 TC SFRUTMIX
45 TC PUTDCSF2
46 TC SFRUTNOR
47
48 PUTDCSF2 INDEX A
49 CAF SFINTABR
```



```
1
2 TC BANKJUMP # SWITCH BANKS FOR EXPANSION ROOM
3 CADR GOALMCYC # ALARM AND RECYCLE IF DEC LOAD
4 # WITH OCTAL ONLY NOUN.
5 CADR BINROUND
6 CADR DEGINSF
7 CADR ARTHINSF
8 CADR DPINSF
9 CADR DPINSF2
10 CADR DSPALARM # LRPOSOUT CANT BE LOADED.
11 CADR DPINSF # SAME AS ARITHDP1
12 CADR HMSIN
13 CADR DSPALARM # MIN/SEC CANT BE LOADED.
14 CADR DPINSF4
15 CADR ARTIN1SF
16 CADR DSPALARM # 2INTOUT CANT BE LOADED.
17 CADR DEGINSF # TESTS AT END FOR 360-CDU
18 ENDRUTIN EQUALS
19
20 # SCALE FACTORS FOR THOSE ROUTINES NEEDING THEM ARE AVAILABLE IN SFTEMP1.
21 # ALL SFIN ROUTINES USE MPAC MPAC+1. LEAVE RESULT IN A. END WITH TC DECRET.
22
23 SETLOC ENDDPDEC +1
24
25 COUNT* $$/PIN
26 # DEGINSF APPLIES 1000/180 = 5.55555(10) = 5.43434(8)
27
28 DEGINSF TC DMP # SF ROUTINE FOR DEC DEGREES
29 ADRES DEGCON1 # MULT BY 5.5 5(10)X2EXP-3
30 CCS MPAC +1 # THIS ROUNDS OFF MPAC+1 BEFORE SHIFT
31 CAF BIT11 # LEFT 3, AND CAUSES 360.00 TO OF/UF
32 TC +2 # WHEN SHIFTED LEFT AND ALARM.
33 CS BIT11
34 AD MPAC +1
35 TC 2ROUND +2
36 TC TPSL1 # LEFT 1
37 DEGINSF2 TC TPSL1 # LEFT 2
38 TC TESTOFUF
39 TC TPSL1 # RETURNS IF NO OF/UF (LEFT3)
40 CCS MPAC
41 TC SIGNFIX # IF +, GO TO SIGNFIX
42 TC SIGNFIX # IF +0, GO TO SIGNFIX
43 COM MPAC # IF -, USE -MAGNITUDE +1
44 TS MPAC # IF -0, USE +0
45 SIGNFIX CCS MPAC+6
46 TC SGNT01 # IF OVERFLOW
47 TC ENDSCALE # NO OVERFLOW/UNDERFLOW
48 CCS MPAC # IF UF FORCE SIGN TO 0 EXCEPT -180
49 TC CCSHOLE
50
51
52
53
54
55
56
57
58
59
60
```

|          |        |                 |                                           |
|----------|--------|-----------------|-------------------------------------------|
|          | TC     | NEG180          |                                           |
|          | TC     | +1              |                                           |
|          | XCH    | MPAC            |                                           |
|          | MASK   | POSMAX          |                                           |
|          | TS     | MPAC            |                                           |
| ENDSCALE | INDEX  | MIXBR           | # IF ROUTINE NO. IS NOT CDU DEGREES,      |
|          | TC     | +0              | # THEN THIS IS 360 - CDU DEGREES          |
|          | TC     | +3              | # AND ANGLE IN MPAC MUST BE REPLACED      |
|          |        |                 | # BY 360 DEGREES MINUS ITSELF.            |
| MIXBACK  | TC     | SFMIXCAL        |                                           |
|          | TC     | +2              |                                           |
|          | TC     | SFNORCAL        |                                           |
| NORBACK  | CS     | A               |                                           |
|          | AD     | BIT2            |                                           |
|          | EXTEND |                 |                                           |
|          | BZF    | +2              |                                           |
|          | TC     | 360-CDU         |                                           |
| ENDSCAL1 | TC     | POSTJUMP        |                                           |
|          | CADR   | PUTCOM2         |                                           |
| SFMIXCAL | TC     | BANKCALL        |                                           |
|          | CADR   | SFRUTMIX        |                                           |
|          | TC     | MIXBACK         |                                           |
| SFNORCAL | TC     | BANKCALL        |                                           |
|          | CADR   | SFRUTNOR        |                                           |
|          | TC     | NORBACK         |                                           |
| NEG180   | CS     | POSMAX          |                                           |
|          | TC     | ENDSCALE -1     |                                           |
| SGNT01   | CS     | MPAC            | # IF OF FORCE SIGN TO 1                   |
|          | MASK   | POSMAX          |                                           |
|          | CS     | A               |                                           |
|          | TC     | ENDSCALE -1     |                                           |
| DEGCON1  | 2DEC   | 5.555555555 B-3 |                                           |
| ARTHINSF | TC     | DMP             | # SCALES MPAC, +1 BY SFTEMP1, SFTEMP2.    |
|          | ADRES  | SFTEMP1         | # ASSUMES POINT BETWEEN HI AND LO PARTS   |
|          | XCH    | MPAC +2         | # OF SFCON. SHIFTS RESULTS LEFT BY 14.    |
|          | XCH    | MPAC +1         | # (BY TAKING RESULTS FROM MPAC+1, MPAC+2) |
|          | XCH    | MPAC            |                                           |
|          | EXTEND |                 |                                           |
|          | BZF    | BINROUND        |                                           |
| BINROUND | TC     | ALMCYCLE        | # TOO LARGE A LOAD. ALARM AND RECYCLE.    |
|          | TC     | 2ROUND          |                                           |
|          | TC     | TESTOFUF        |                                           |
|          | TC     | ENDSCAL1        | # RETURNS IF NO OF/UF                     |

|          |        |           |                                            |
|----------|--------|-----------|--------------------------------------------|
| ARTIN1SF | TC     | DMP       | # SCALES MPAC, +1 BY SFTEMP1, SFTEMP2.     |
|          | ADRES  | SFTEMP1   | # ROUNDS MPAC+1 INTO MPAC.                 |
|          | TC     | BINROUND  |                                            |
| DPINSF   | TC     | DMP       | # SCALES MPAC, MPAC +1 BY SFTEMP1,         |
|          | ADRES  | SFTEMP1   | # SFTEMP. STORES LOW PART OF RESULT        |
|          | XCH    | MPAC +2   | # IN (E SUBK) +1 OR E+1                    |
|          | DOUBLE |           |                                            |
|          | TS     | MPAC +2   |                                            |
|          | CAF    | ZERO      |                                            |
|          | AD     | MPAC +1   |                                            |
|          | TC     | 2ROUND +2 |                                            |
|          | TC     | TESTOFUF  |                                            |
|          | INDEX  | MIXBR     | # RETURNS IF NO OF/UF                      |
|          | TC     | +0        |                                            |
|          | TC     | DPINORM   |                                            |
|          | CA     | DECOUNT   | # MIXED NOUN                               |
| DPINCOM  | AD     | NOUNADD   | # MIXED NORMAL                             |
|          | TS     | Q         | # E SUBK E                                 |
|          | XCH    | MPAC +1   |                                            |
|          | INDEX  | Q         |                                            |
|          | TS     | 1         | # PLACE LOW PART IN                        |
|          | TC     | ENDSCAL1  | # (E SUBK) +1 MIXED                        |
| DPINORM  | CAF    | ZERO      | # E +1 NORMAL                              |
|          | TC     | DPINCOM   |                                            |
| DPINSF2  | TC     | DMP       | # ASSUMES POINT BETWEEN BITS 7-8 OF HIGH   |
|          | ADRES  | SFTEMP1   | # PART OF SF CONST. DPINSF2 SHIFTS RESULTS |
|          | CAF    | SIX       | # LEFT BY 7, ROUNDS MPAC+2 INTO MPAC+1     |
|          | TC     | TPLEFTN   | # SHIFT LEFT 7.                            |
|          | TC     | DPINSF +2 |                                            |
| DPINSF4  | TC     | DMP       | # ASSUMES POINT BETWEEN BITS 11-12 OF HIGH |
|          | ADRES  | SFTEMP1   | # PART OF SF CONST. DPINSF2 SHIFTS RESULTS |
|          | CAF    | TWO       | # LEFT BY 3, ROUNDS MPAC+2 INTO MPAC+1.    |
|          | TC     | TPLEFTN   | # SHIFT LEFT 3.                            |
|          | TC     | DPINSF +2 |                                            |
| TPLEFTN  | XCH    | Q         | # SHIFTS MPAC, +1, +2 LEFT N. SETS OVFIN   |
|          | TS     | SFTEMP2   | # TO +1 FOR OF, -1 FOR UF.                 |
| LEFTNCOM | XCH    | Q         | # CALL WITH N-1 IN A.                      |
|          | TS     | SFTEMP1   | # LOOP TIME .37 MSEC.                      |
|          | TC     | TPSL1     |                                            |
|          | CCS    | SFTEMP1   |                                            |
|          | TC     | LEFTNCOM  |                                            |

|    |          |        |          |                                          |
|----|----------|--------|----------|------------------------------------------|
| 1  |          | TC     | SFTEMP2  |                                          |
| 2  |          |        |          |                                          |
| 3  |          |        |          |                                          |
| 4  | 2ROUND   | XCH    | MPAC     | +1                                       |
| 5  |          | DOUBLE |          |                                          |
| 6  |          | TS     | MPAC     | +1                                       |
| 7  |          | TC     | Q        | # IF MPAC+1 DOES NOT OF/UF               |
| 8  |          | AD     | MPAC     |                                          |
| 9  |          | TS     | MPAC     |                                          |
| 10 |          | TC     | Q        | # IF MPAC DOES NOT OF/UF                 |
| 11 |          | TS     | MPAC+6   |                                          |
| 12 | 2RNDEND  | TC     | Q        |                                          |
| 13 |          |        |          |                                          |
| 14 | TESTOFUF | CCS    | MPAC+6   | # RETURNS IF NO OF/UF                    |
| 15 |          | TC     | ALMCYCLE | # OF ALARM AND RECYCLE.                  |
| 16 |          | TC     | Q        |                                          |
| 17 |          | TC     | ALMCYCLE | # UF ALARM AND RECYCLE.                  |
| 18 |          |        |          |                                          |
| 19 |          | SETLOC | ENDSPMIN | +1                                       |
| 20 |          |        |          |                                          |
| 21 |          | COUNT* | \$\$/PIN |                                          |
| 22 | HMSIN    | TC     | ALL3DEC  | # IF ALL 3 WORDS WERE NOT LOADED, ALARM. |
| 23 |          | TC     | DMP      | # XREG, XREGLP (=HOURS) WERE ALREADY PUT |
| 24 |          | ADRES  | WHOLECON | # INTO MPAC, MPAC+1.                     |
| 25 |          | TC     | RND/TST  | # ROUND OFF TO WHOLE HRS IN MPAC+1.      |
| 26 |          | CAF    | ZERO     | # ALARM IF MPAC NON ZERO (G/ 16383).     |
| 27 |          | TS     | MPAC     | +2                                       |
| 28 |          | CAF    | HRCON    |                                          |
| 29 |          | TS     | MPAC     |                                          |
| 30 |          | CAF    | HRCON    | +1                                       |
| 31 |          | XCH    | MPAC     | +1                                       |
| 32 |          | TC     | SHORTMP  |                                          |
| 33 |          | TC     | MPACTST  | # ALARM IF MPAC NON ZERO (G/ 745)        |
| 34 |          | DXCH   | MPAC     | +1                                       |
| 35 |          | DXCH   | HITEMIN  | # STORE HOURS CONTRIBUTION               |
| 36 |          | CA     | YREG     | # PUT YREG, YREGLP INTO MPAC, +1.        |
| 37 |          | LXCH   | YREGLP   |                                          |
| 38 |          | DXCH   | MPAC     |                                          |
| 39 |          | TC     | DMP      |                                          |
| 40 |          | ADRES  | WHOLECON |                                          |
| 41 |          | TC     | RND/TST  | # ROUND OFF TO WHOLE MIN IN MPAC+1       |
| 42 |          | CS     | 59MIN    | # ALARM IF MPAC NON ZERO (G/16383)       |
| 43 |          | TC     | SIZETST  | # ALARM IF MPAC+1 G/ 59MIN               |
| 44 |          | XCH    | MPAC     | +1                                       |
| 45 |          | EXTEND |          |                                          |
| 46 |          | MP     | MINCON   | # LEAVES MINUTES CONTRIBUTION IN A,L     |
| 47 |          | DAS    | HITEMIN  | # ADD IN MINUTES CONTRIBUTION            |
| 48 |          | EXTEND |          | # IF THIS DAS OVEFLOWS, G/ 745 HR, 39MIN |
| 49 |          |        |          |                                          |
| 50 |          |        |          |                                          |
| 51 |          |        |          |                                          |
| 52 |          |        |          |                                          |
| 53 |          |        |          |                                          |
| 54 |          |        |          |                                          |
| 55 |          |        |          |                                          |
| 56 |          |        |          |                                          |
| 57 |          |        |          |                                          |
| 58 |          |        |          |                                          |
| 59 |          |        |          |                                          |
| 60 |          |        |          |                                          |

|    |          |        |          |                                          |
|----|----------|--------|----------|------------------------------------------|
| 1  |          |        |          |                                          |
| 2  |          | BZF    | +2       |                                          |
| 3  |          | TC     | ALMCYCLE |                                          |
| 4  |          | CA     | ZREG     | # PUT ZREG, ZREGLP INTO MPAC, +1.        |
| 5  |          | LXCH   | ZREGLP   |                                          |
| 6  |          | DXCH   | MPAC     |                                          |
| 7  |          | TC     | DMP      |                                          |
| 8  |          | ADRES  | WHOLECON |                                          |
| 9  |          | TC     | RND/TST  | # ROUND OFF TO WHOLE CENTI-SEC IN MPAC+1 |
| 10 |          | CS     | 59.99SEC | # ALARM IF MPAC NON ZERO (G/163.83 SEC)  |
| 11 |          | TC     | SIZETST  | # ALARM IF MPAC+1 G/59.99 SEC            |
| 12 |          | DXCH   | HITEMIN  | # ADD IN SECONDS CONTRIBUTION            |
| 13 |          | DAS    | MPAC     | # IF THIS DAS OVERFLOWS,                 |
| 14 |          | EXTEND |          | # G/ 745 HR, 39 MIN, 14.59 SEC.          |
| 15 |          | BZF    | +2       |                                          |
| 16 |          | TC     | ALMCYCLE | # ALARM AND RECYCLE                      |
| 17 |          | CAF    | ZERO     |                                          |
| 18 |          | TS     | MPAC +2  |                                          |
| 19 |          | TC     | TPAGREE  |                                          |
| 20 |          | DXCH   | MPAC     |                                          |
| 21 |          | INDEX  | NOUNADD  |                                          |
| 22 |          | DXCH   | 0        |                                          |
| 23 |          | TC     | POSTJUMP |                                          |
| 24 |          | CADR   | LOADLV   |                                          |
| 25 |          |        |          |                                          |
| 26 | WHOLECON | OCT    | 00006    | # (10EXP5/2EXP14)2EXP14                  |
| 27 |          | OCT    | 03240    |                                          |
| 28 | HRCON    | OCT    | 00025    | # 1 HOUR IN CENTI-SEC                    |
| 29 |          | OCT    | 37100    |                                          |
| 30 | MINCON   | OCT    | 13560    | # 1 MINUTE IN CENTI-SEC                  |
| 31 | 59MIN    | OCT    | 00073    | # 59 AS WHOLE                            |
| 32 | 59.99SEC | OCT    | 13557    | # 5999 CENTI-SEC                         |
| 33 |          |        |          |                                          |
| 34 | RND/TST  | XCH    | MPAC +2  | # ROUNDS MPAC+2 INTO MPAC+1.             |
| 35 |          | DOUBLE |          | # ALARMS IF MPAC NOT 0                   |
| 36 |          | TS     | MPAC +2  |                                          |
| 37 |          | CAF    | ZERO     |                                          |
| 38 |          | AD     | MPAC +1  |                                          |
| 39 |          | TS     | MPAC +1  |                                          |
| 40 |          | CAF    | ZERO     |                                          |
| 41 |          | AD     | MPAC     | # CANT OVFLOW                            |
| 42 |          | XCH    | MPAC     |                                          |
| 43 | MPACTST  | CCS    | MPAC     | # ALARM IF MPAC NON ZERO                 |
| 44 |          | TC     | ALMCYCLE | # ALARM AND RECYCLE.                     |
| 45 |          | TC     | Q        |                                          |
| 46 |          | TC     | ALMCYCLE | # ALARM AND RECYCLE.                     |
| 47 |          | TC     | Q        |                                          |
| 48 |          |        |          |                                          |
| 49 | SIZETST  | TS     | MPAC +2  | # CALLED WITH - CON IN A                 |
| 50 |          | CCS    | MPAC +1  | # GET MAG OF MPAC+1                      |
| 51 |          |        |          |                                          |
| 52 |          |        |          |                                          |
| 53 |          |        |          |                                          |
| 54 |          |        |          |                                          |
| 55 |          |        |          |                                          |
| 56 |          |        |          |                                          |
| 57 |          |        |          |                                          |
| 58 |          |        |          |                                          |
| 59 |          |        |          |                                          |
| 60 |          |        |          |                                          |

|    |                                                                 |        |          |                                            |    |
|----|-----------------------------------------------------------------|--------|----------|--------------------------------------------|----|
| 1  |                                                                 |        |          |                                            | 1  |
| 2  |                                                                 | AD     | ONE      |                                            | 2  |
| 3  |                                                                 | TCF    | +2       |                                            | 3  |
| 4  |                                                                 | AD     | ONE      |                                            | 4  |
| 5  |                                                                 | AD     | MPAC +2  |                                            | 5  |
| 6  |                                                                 | EXTEND |          | # MAG OF MPAC+1 - CON                      | 6  |
| 7  |                                                                 | BZMF   | +2       |                                            | 7  |
| 8  |                                                                 | TC     | ALMCYCLE | # MAG OF MPAC+1 G/ CON. ALARM AND RECYCLE. | 8  |
| 9  |                                                                 | TC     | Q        | # MAG OF MPAC+1 L/= CON                    | 9  |
| 10 |                                                                 |        |          |                                            | 10 |
| 11 | # ALL3DEC TESTS THAT ALL 3 WORDS ARE LOADED IN DEC (FOR HMSIN). |        |          |                                            | 11 |
| 12 | # ALARM IF NOT. (TEST THAT BITS 3,4,5 OF DECBRNCH ARE ALL = 1). |        |          |                                            | 12 |
| 13 |                                                                 |        |          |                                            | 13 |
| 14 | ALL3DEC                                                         | CS     | OCT34BAR | # GET BITS 3,4,5 IN A                      | 14 |
| 15 |                                                                 | MASK   | DECBRNCH | # GET BITS 3,4,5 OF DECBRNCH IN A          | 15 |
| 16 |                                                                 | AD     | OCT34BAR | # BITS 3,4,5 OF DECBRNCH MUST ALL = 1      | 16 |
| 17 |                                                                 | CCS    | A        |                                            | 17 |
| 18 |                                                                 | TC     | FORCEV25 |                                            | 18 |
| 19 | OCT34BAR                                                        | OCT    | 77743    |                                            | 19 |
| 20 |                                                                 | TC     | FORCEV25 |                                            | 20 |
| 21 |                                                                 | TC     | Q        |                                            | 21 |
| 22 |                                                                 |        |          |                                            | 22 |
| 23 | FORCEV25                                                        | CS     | OCT31    | # FORCE VERB 25 TO BE EXECUTED BY RECYCLE  | 23 |
| 24 |                                                                 | TS     | VERBSAVE | # IN CASE OPERATOR EXECUTED A LOWER LOAD   | 24 |
| 25 |                                                                 | TC     | ALMCYCLE | # VERB. ALARM AND RECYCLE.                 | 25 |
| 26 | ENDHMSS                                                         | EQUALS |          |                                            | 26 |
| 27 |                                                                 |        |          |                                            | 27 |
| 28 |                                                                 |        |          |                                            | 28 |
| 29 |                                                                 |        |          |                                            | 29 |
| 30 |                                                                 |        |          |                                            | 30 |
| 31 |                                                                 |        |          |                                            | 31 |
| 32 |                                                                 |        |          |                                            | 32 |
| 33 |                                                                 |        |          |                                            | 33 |
| 34 |                                                                 |        |          |                                            | 34 |
| 35 |                                                                 |        |          |                                            | 35 |
| 36 |                                                                 |        |          |                                            | 36 |
| 37 |                                                                 |        |          |                                            | 37 |
| 38 |                                                                 |        |          |                                            | 38 |
| 39 |                                                                 |        |          |                                            | 39 |
| 40 |                                                                 |        |          |                                            | 40 |
| 41 |                                                                 |        |          |                                            | 41 |
| 42 |                                                                 |        |          |                                            | 42 |
| 43 |                                                                 |        |          |                                            | 43 |
| 44 |                                                                 |        |          |                                            | 44 |
| 45 |                                                                 |        |          |                                            | 45 |
| 46 |                                                                 |        |          |                                            | 46 |
| 47 |                                                                 |        |          |                                            | 47 |
| 48 |                                                                 |        |          |                                            | 48 |
| 49 |                                                                 |        |          |                                            | 49 |
| 50 |                                                                 |        |          |                                            | 50 |
| 51 |                                                                 |        |          |                                            | 51 |
| 52 |                                                                 |        |          |                                            | 52 |
| 53 |                                                                 |        |          |                                            | 53 |
| 54 |                                                                 |        |          |                                            | 54 |
| 55 |                                                                 |        |          |                                            | 55 |
| 56 |                                                                 |        |          |                                            | 56 |
| 57 |                                                                 |        |          |                                            | 57 |
| 58 |                                                                 |        |          |                                            | 58 |
| 59 |                                                                 |        |          |                                            | 59 |
| 60 |                                                                 |        |          |                                            | 60 |

```
1 # MONITOR ALLOWS OTHER KEYBOARD ACTIVITY. IT IS ENDED BY VERB TERMINATE,
2 # VERB PROCEED WITHOUT DATA, VERB RESEQUENCE,
3 # ANOTHER MONITOR, OR ANY NVSUB CALL THAT PASSES THE DSPLOCK (PROVIDED
4 # THAT THE OPERATOR HAS SOMEHOW ALLOWED THE ENDING OF A MONITOR WHICH
5 # HE HAS INITIATED THROUGH THE KEYBOARD).
6 #
7 # MONITOR ACTION IS SUSPENDED, BUT NOT ENDED, BY ANY KEYBOARD ACTION,
8 # EXCEPT ERROR LIGHT RESET. IT BEGINS AGAIN WHEN KEY RELEASE IS PERFORMED.
9 # MONITOR SAVES THE NOUN AND APPROPRIATE DISPLAY VERB IN MONSAVE. IT SAVES
10 # NOUNCADR IN MONSAVE1, IF NOUN = MACHINE CADR TO BE SPECIFIED. BIT 15 OF
11 # MONSAVE1 IS THE KILL MONITOR SIGNAL (KILLER BIT). BIT 14 OF MONSAVE1
12 # INDICATES THE CURRENT MONITOR WAS EXTERNALLY INITIATED (EXTERNAL
13 # MONITOR BIT). IT IS TURNED OFF BY RELDSP AND KILMONON.
14 #
15 # MONSAVE INDICATES IF MONITOR IS ON (+=ON, +0=OFF)
16 # IF MONSAVE IS +, MONITOR ENTERS NO REQUEST, BUT TURNS KILLER BIT OFF.
17 # IF MONSAVE IS +0, MONITOR ENTERS REQUEST AND TURNS KILLER BIT OFF.
18 #
19 # NVSUB (IF EXTERNAL MONITOR BIT IS OFF), VB=PROCEED WITHOUT DATA,
20 # VB=RESEQUENCE, AND VB=TERMINATE TURN KILL MONITOR BIT ON.
21 #
22 # IF KILLER BIT IS ON, MONREQ ENTERS NO FURTHER REQUESTS, ZEROS MONSAVE
23 # AND MONSAVE1 (TURNING OFF KILLER BIT AND EXTERNAL MONITOR BIT).
24 #
25 # MONITOR DOESNT TEST FOR MATBS SINCE NVSUB CAN HANDLE INTERNAL MATBS NOW.
```

```
28 SETLOC ENDRUTIN
```

```
30 COUNT* $$/PIN
```

```
31 MONITOR CS BIT15/14
32 MASK NOUNCADR
33 MONIT1 TS MPAC +1 # TEMP STORAGE
```

```
34 CS ENTEXT
35 AD ENDINST
36 CCS A
```

```
37 BIT15/14 TC MONIT2
38 OCT 60000
39 TC MONIT2
```

```
40 CAF BIT14 # EXTERNALLY INITIATED MONITOR.
41 ADS MPAC +1 # SET BIT 14 FOR MONSAVE1.
42 CAF ZERO
```

```
43 MONIT2 TS MONSAVE2 # ZERO NVMONOPT OPTIONS
44 CAF LOW7
45 MASK VERBREG
```

```
46 TC LEFT5
47 TS CYL
48 CS CYL
```

```
49 XCH CYL
50 AD NOUNREG
51 TS MPAC # TEMP STORAGE
```

```
52 CAF ZERO
```

|    |         |        |          |                                            |
|----|---------|--------|----------|--------------------------------------------|
| 1  |         |        |          |                                            |
| 2  |         | TS     | DSPLOCK  | # +0 INTO DSPLOCK SO MONITOR CAN RUN.      |
| 3  |         | CCS    | CADRSTOR | # TURN OFF KR LITE IF CADRSTOR AND DSPLIST |
| 4  |         | TC     | +2       | # ARE BOTH EMPTY. (LITE COMES ON IF NEW    |
| 5  |         | TC     | RELDSP1  | # MONITOR IS KEYED IN OVER OLD MONITOR.)   |
| 6  |         | INHINT |          |                                            |
| 7  |         | CCS    | MONSAVE  |                                            |
| 8  |         | TC     | +5       | # IF MONSAVE WAS +, NO REQUEST             |
| 9  |         | CAF    | ONE      | # IF MONSAVE WAS 0, REQUEST MONREQ         |
| 10 |         | TC     | WAITLIST |                                            |
| 11 |         | EBANK= | DSPCOUNT |                                            |
| 12 |         | 2CADR  | MONREQ   |                                            |
| 13 |         |        |          |                                            |
| 14 |         | DXCH   | MPAC     | # PLACE MONITOR VERB AND NOUN INTO MONSAVE |
| 15 |         | DXCH   | MONSAVE  | # ZERO THE KILL MONITOR BIT                |
| 16 |         | RELINT |          | # SET UP EXTERNAL MONITOR BIT              |
| 17 |         | TC     | ENTRET   |                                            |
| 18 |         |        |          |                                            |
| 19 | MONREQ  | TC     | LODSAMPT | # CALLED BY WAITLIST                       |
| 20 |         | CCS    | MONSAVE1 | # TIME IS SNATCHED N RUPT FOR NOUN 65      |
| 21 |         | TC     | +4       | # IF KILLER BIT = 0, ENTER REQUESTS        |
| 22 |         | TC     | +3       | # IF KILLER BIT = 0, ENTER REQUESTS        |
| 23 |         | TC     | KILLMON  | # IF KILLER BIT = 1, NO REQUESTS.          |
| 24 |         | TC     | KILLMON  | # IF KILLER BIT = 1, NO REQUESTS.          |
| 25 |         | CAF    | MONDEL   |                                            |
| 26 |         | TC     | WAITLIST | # ENTER WAITLIST REQUEST FOR MONREQ        |
| 27 |         | EBANK= | DSPCOUNT |                                            |
| 28 |         | 2CADR  | MONREQ   |                                            |
| 29 |         |        |          |                                            |
| 30 |         | CAF    | CHRPRI0  |                                            |
| 31 |         | TC     | NOVAC    | # ENTER EXEC REQUEST FOR MONDO             |
| 32 |         | EBANK= | DSPCOUNT |                                            |
| 33 |         | 2CADR  | MONDO    |                                            |
| 34 |         |        |          |                                            |
| 35 |         | TC     | TASKOVER |                                            |
| 36 |         |        |          |                                            |
| 37 | KILLMON | CAF    | ZERO     | # ZERO MONSAVE AND TURN KILLER BIT OFF     |
| 38 |         | TS     | MONSAVE  |                                            |
| 39 |         | TS     | MONSAVE1 | # TURN OFF KILL MONITOR BIT.               |
| 40 |         | TC     | TASKOVER | # TURN OFF EXTERNAL MONITOR BIT.           |
| 41 | MONDEL  | OCT    | 144      | # FOR 1 SEC MONITOR INTERVALS              |
| 42 |         |        |          |                                            |
| 43 | MONDO   | CCS    | MONSAVE1 | # CALLED BY EXEC                           |
| 44 |         | TC     | +4       | # IF KILLER BIT = 0, CONTINUE              |
| 45 |         | TC     | +3       | # IF KILLER BIT = 0, CONTINUE              |
| 46 |         | TC     | ENDOFJOB | # IN CASE TERMINATE CAME SINCE LAST MONREQ |
| 47 |         | TC     | ENDOFJOB | # IN CASE TERMINATE CAME SINCE LAST MONREQ |
| 48 |         | CCS    | DSPLOCK  |                                            |
| 49 |         | TC     | MONBUSY  | # NVSUB IS BUSY                            |
| 50 |         |        |          |                                            |
| 51 |         |        |          |                                            |
| 52 |         |        |          |                                            |
| 53 |         |        |          |                                            |
| 54 |         |        |          |                                            |
| 55 |         |        |          |                                            |
| 56 |         |        |          |                                            |
| 57 |         |        |          |                                            |
| 58 |         |        |          |                                            |
| 59 |         |        |          |                                            |
| 60 |         |        |          |                                            |



|    |          |        |             |                                            |
|----|----------|--------|-------------|--------------------------------------------|
| 1  |          |        |             |                                            |
| 2  |          | CAF    | LOW7        |                                            |
| 3  |          | MASK   | MONSAVE     |                                            |
| 4  |          | TC     | UPDATNN -1  | # PLACE NOUN INTO NOUNREG AND DISPLAY IT   |
| 5  |          | CAF    | MID7        |                                            |
| 6  |          | MASK   | MONSAVE     | # CHANGE MONITOR VERB TO DISPLAY VERB      |
| 7  |          | AD     | MONREF      | # -DEC10, STARTING IN BIT8                 |
| 8  |          | TS     | EDOP        | # RIGHT 7                                  |
| 9  |          | CA     | EDOP        |                                            |
| 10 |          | TS     | VERBREG     |                                            |
| 11 |          | CAF    | MONBACK     | # SET RETURN TO PASTEVB AFTER DATA DISPLAY |
| 12 |          | TS     | ENTRET      |                                            |
| 13 |          | CS     | BIT15/14    |                                            |
| 14 |          | MASK   | MONSAVE1    | # PUT ECADR INTO MPAC +2. INTMCTBS WILL    |
| 15 |          | TS     | MPAC +2     | # DISPLAY IT AND SET NOUNCADR, NOUNADD,    |
| 16 | ENDMONDO | TC     | TESTNN      | # EBANK.                                   |
| 17 |          |        |             |                                            |
| 18 |          | BLOCK  | 2           |                                            |
| 19 |          |        |             |                                            |
| 20 |          | SETLOC | FFTAG8      |                                            |
| 21 |          | BANK   |             |                                            |
| 22 |          |        |             |                                            |
| 23 |          | COUNT* | \$\$/PIN    |                                            |
| 24 | PASTEVB  | CAF    | MID7        |                                            |
| 25 |          | MASK   | MONSAVE2    | # NVMONOPT PASTE OPTION                    |
| 26 |          | EXTEND |             |                                            |
| 27 |          | BZF    | +2          |                                            |
| 28 |          | TC     | PASTEOPT    | # PASTE PLEASE VERB FOR NVMONOPT           |
| 29 |          | CA     | MONSAVE     | # PASTE MONITOR VERB - PASTE OPTION IS 0   |
| 30 | PASTEOPT | TS     | EDOP        | # RIGHT 7                                  |
| 31 |          | CA     | EDOP        | # PLACE MONITOR VERB OR PLEASE VERB INTO   |
| 32 |          | TC     | BANKCALL    | # VERBREG AND DISPLAY IT.                  |
| 33 |          | CADR   | UPDATVB -1  |                                            |
| 34 |          | CAF    | ZERO        | # ZERO REQRET SO THAT PASTED VERBS CAN     |
| 35 |          | TS     | REQRET      | # BE EXECUTED BY OPERATOR.                 |
| 36 |          | CA     | MONSAVE2    |                                            |
| 37 |          | TC     | BLANKSUB    | # PROCESS NVMONOPT BLANK OPTION IF ANY     |
| 38 |          | TC     | +1          |                                            |
| 39 | ENDPASTE | TC     | ENDOFJOB    |                                            |
| 40 |          |        |             |                                            |
| 41 | MID7     | OCT    | 37600       |                                            |
| 42 |          |        |             |                                            |
| 43 |          | SETLOC | ENDMONDO +1 |                                            |
| 44 |          | COUNT* | \$\$/PIN    |                                            |
| 45 | MONREF   | OCT    | 75377       | # -DEC10, STARTING IN BIT8                 |
| 46 | MONBACK  | ADRES  | PASTEVB     |                                            |
| 47 |          |        |             |                                            |
| 48 | MONBUSY  | TC     | RELDSPON    | # TURN KEY RELEASE LIGHT                   |
| 49 |          | TC     | ENDOFJOB    |                                            |
| 50 |          |        |             |                                            |
| 51 |          |        |             |                                            |
| 52 |          |        |             |                                            |
| 53 |          |        |             |                                            |
| 54 |          |        |             |                                            |
| 55 |          |        |             |                                            |
| 56 |          |        |             |                                            |
| 57 |          |        |             |                                            |
| 58 |          |        |             |                                            |
| 59 |          |        |             |                                            |
| 60 |          |        |             |                                            |

# PINBALL\_GAME\_BUTTONS\_AND\_LIGHTS

```
DSPFMEM IS USED TO DISPLAY (IN OCTAL) ANY FIXED REGISTER.
IT IS USED WITH NOUN = MACHINE CADR TO BE SPECIFIED. THE FCADR OF THE
DESIRED LOCATION IS THEN PUNCHED IN. IT HANDLES F/F (FCADR 4000-7777)
#
FOR BANKS L/E 27, THIS IS ENOUGH.
#
FOR BANKS G/E 30, THE THIRD COMPONENT OF NOUN 26 (PRIO, ADRES, BBCON)
MUST BE PRELOADED WITH THE DESIRED SUPERBANK BITS (BITS 5,6,7).
V23N26 SHOULD BE USED.
#
SUMMARY
FOR BANKS L/E 27, V27N01E(FCADR)E
FOR BANKS G/E 30, V23N26E(SUPERBITS)E V27N01E(FCADR)E

DSPFMEM CAF R1D1 # IF F/F, DATACALL USES BANK 02 OR 03.
 TS DSPCOUNT
 CA DSPTM1 +2 # SUPERBANK BITS WERE PRELOADED INTO
 TS L # 3RD COMPONENT OF NOUN 26.
 CA NOUNCADR # ORIGINAL FCADR LOADED STILL IN NOUNCADR.
 TC SUPDACAL # CALL WITH FCADR IN A, SUPERBITS IN L.

ENDSPF TC DSPOCTWO
 TC ENDOFJOB
```

1412THE

## # WORD DISPLAY ROUTINES

SETLOC TESTOFUF +4

COUNT\* \$\$/PIN

DPSIGN XCH Q

TS DSPWDRET

CCS MPAC

TC +8D

TC +7

AD ONE

TS MPAC

TC -ON

CS MPAC +1

TS MPAC +1

TC DSPWDRET

TC +ON

TC DSPWDRET

DSPRND EXTEND # ROUND BY 5 EXP-6

DCA DECROUND -1

DAS MPAC

EXTEND

BZF +4

EXTEND

DCA DPOSMAX

DXCH MPAC

TC Q

# DSPDECWD CONVERTS C(MPAC, MPAC+1) INTO A SIGN AND 5 CHAR DECIMAL

# STARTING IN LOC SPECIFIED IN DSPCOUNT. IT ROUNDS BY 5 EXP-6.

DSPDECWD XCH Q

TS WDRET

TC DSPSIGN

TC DSPRND

CAF FOUR

DSPDCWD1 TS WDCNT

CAF BINCON

TC SHORTMP

TRACE1 INDEX MPAC

CAF RELTAB

MASK LOW5

TS CODE

CAF ZERO

XCH MPAC +2

XCH MPAC +1

TS MPAC

XCH DSPCOUNT

TRACE1S TS COUNT

CCS A # DECREMENT DSPCOUNT EXCEPT AT +0

```
1
2 TS DSPCOUNT
3 TC DSPIN
4 CCS WDCNT
5 TC DSPDCWD1
6 CS VD1
7 TS DSPCOUNT
8 TC WDRET
9
10 DECROUND OCT 00000
11 OCT 02476
12
13 # DSPDECNR CONVERTS C(MPAC,MPAC+1) INTO A SIGN AND 5 CHAR DECIMAL
14 # STARTING IN LOC SPECIFIED IN DSPCOUNT. IT DOES NOT ROUND
15
16 DSPDECNR XCH Q
17 TS WDRET
18 TC DSPSIGN
19 TC DSPDCWD1 -1
20
21 # DSPDC2NR CONVERTS C(MPAC,MPAC+1) INTO A SIGN AND 2 CHAR DECIMAL
22 # STARTING IN LOC SPECIFIED IN DSPCOUNT. IT DOES NOT ROUND
23
24 DSPDC2NR XCH Q
25 TS WDRET
26 TC DSPSIGN
27 CAF ONE
28 TC DSPDCWD1
29
30 # DSP2DEC CONVERTS C(MPAC) AND C(MPAC+1) INTO A SIGN AND 10 CHAR DECIMAL
31 # STARTING IN THE LOC SPECIFIED IN DSPCOUNT.
32
33 DSP2DEC XCH Q
34 TS WDRET
35 CAF ZERO
36 TS CODE
37 CAF THREE
38 TC 11DSPIN # -R2 OFF
39 CAF FOUR
40 TC 11DSPIN # +R2 OFF
41 TC DSPSIGN
42 CAF R2D1
43 END2DEC TC DSPDCWD1
44
45 # DSPDECVN DISPLAYS C(A) UPON ENTRY AS A 2 CHAR DECIMAL BEGINNING IN THE
46 # DSP LOC SPECIFIED IN DSPCOUNT.
47 # C(A) SHOULD BE IN FORM N X 2EXP-14. THIS IS SCALED TO FORM N/100 BEFORE
48 # DISPLAY CONVERSION.
```

|          |                                                                          |             |                                            |
|----------|--------------------------------------------------------------------------|-------------|--------------------------------------------|
| DSPDECVN | EXTEND                                                                   |             |                                            |
|          | MP                                                                       | VNDSPCON    | # MULT BY .01                              |
|          | LXCH                                                                     | MPAC        | # TAKE RESULTS FROM L. (MULT BY 2EXP14).   |
|          | CAF                                                                      | ZERO        |                                            |
|          | TS                                                                       | MPAC +1     |                                            |
|          | XCH                                                                      | Q           |                                            |
|          | TS                                                                       | WDRET       |                                            |
|          | TC                                                                       | DSPDC2NR +3 | # NO SIGN, NO ROUND, 2 CHAR                |
| VNDSPCON | OCT                                                                      | 00244       | # .01 ROUNDED UP                           |
| GOVNUPDT | TC                                                                       | DSPDECVN    | # THIS IS NOT FOR GENERAL USE. REALLY PART |
|          | TC                                                                       | POSTJUMP    | # OF UPDATVB.                              |
|          | CADR                                                                     | UPDAT1 +2   |                                            |
| ENDECVN  | EQUALS                                                                   |             |                                            |
|          | SETLOC                                                                   | ENDSPF +1   |                                            |
|          | COUNT*                                                                   | \$\$/PIN    |                                            |
|          | # DSPOCTWD DISPLAYS C(A) UPON ENTRY AS A 5 CHAR OCT STARTING IN THE DSP  |             |                                            |
|          | # CHAR SPECIFIED IN DSPCOUNT. IT STOPS AFTER 5 CHAR HAVE BEEN DISPLAYED. |             |                                            |
| DSPOCTWO | TS                                                                       | CYL         |                                            |
|          | XCH                                                                      | Q           |                                            |
|          | TS                                                                       | WDRET       | # MUST USE SAME RETURN AS DSP2BIT.         |
|          | CAF                                                                      | BIT14       | # TO BLANK SIGNS                           |
|          | ADS                                                                      | DSPCOUNT    |                                            |
|          | CAF                                                                      | FOUR        |                                            |
| WDAGAIN  | TS                                                                       | WDCNT       |                                            |
|          | CS                                                                       | CYL         |                                            |
|          | CS                                                                       | CYL         |                                            |
|          | CS                                                                       | CYL         |                                            |
|          | CS                                                                       | A           |                                            |
|          | MASK                                                                     | DSPMSK      |                                            |
|          | INDEX                                                                    | A           |                                            |
|          | CAF                                                                      | RELTAB      |                                            |
|          | MASK                                                                     | LOW5        |                                            |
|          | TS                                                                       | CODE        |                                            |
|          | XCH                                                                      | DSPCOUNT    |                                            |
|          | TS                                                                       | COUNT       |                                            |
|          | CCS                                                                      | A           | # DECREMENT DSPCOUNT EXCEPT AT +0          |
|          | TS                                                                       | DSPCOUNT    |                                            |
|          | TC                                                                       | POSTJUMP    |                                            |
|          | CADR                                                                     | DSPOCTIN    |                                            |
| OCTBACK  | CCS                                                                      | WDCNT       |                                            |
|          | TC                                                                       | WDAGAIN     | # +                                        |
| DSPLV    | CS                                                                       | VD1         | # TO BLOCK NUMERICAL CHARACTERS, CLEARS,   |
|          | TS                                                                       | DSPCOUNT    | # AND SIGNS AFTER A COMPLETED DISPLAY.     |

TC WDRET

DSPMSK = SEVEN

# DSP2BIT DISPLAYS C(A) UPON ENTRY AS A 2 CHAR OCT BEGINNING IN THE DSP  
# LOC SPECIFIED IN DSPCOUNT BY PRE CYCLING RIGHT C(A) AND USING THE LOGIC  
# OF THE 5 CHAR OCTAL DISPLAY

DSP2BIT TS CYR  
XCH Q  
TS WDRET

CAF ONE  
TS WDCNT  
CS CYR

CS CYR  
XCH CYR  
TS CYL

TC WDAGAIN +5

# FOR DSPIN PLACE 0/25 OCT INTO COUNT, 5 BIT RELAY CODE INTO CODE. BOTH  
# ARE DESTROYED. IF BIT14 OF COUNT IS 1, SIGN IS BLANKED WITH LEFT CHAR.  
# FOR DSPIN1 PLACE 0,1 INTO BIT11 OF CODE, 2 INTO COUNT, REL ADDRESS OF  
# DSPTAB ENTRY INTO DSREL.

SETLOC ENDECVN

DSPIN COUNT\* \$\$/PIN  
XCH Q # CANT USE L FOR RETURN, SINCE MANYOF THE  
TS DSEXIT # ROTINE CALLING DSPIN USE L AS RETURN.

CAF LOW5  
MASK COUNT  
TS SR

XCH SR  
TS DSREL  
CAF BIT1

MASK COUNT  
CCS A  
TC +2

TC DSPIN1 -1 # LEFT IF COUNT IS ODD  
XCH CODE # RIGHT IF COUNT IS EVEN  
TC SLEFT5 # DOES NOT USE CYL

TS CODE  
CAF BIT14  
MASK COUNT

CCS A  
CAF TWO # BIT14 = 1, BLANK SIGN  
AD ONE # BIT14 = 0, LEAVE SIGN ALONE  
TS COUNT # +0 INTO COUNT FOR RIGHT

# +1 INTO COUNT FOR LEFT (SIGN LEFT ALONE)  
# +3 INTO COUNT FOR LEFT (TO BLANK SIGN)

DSPIN1

INHINT

INDEX

CCS

DSREL

DSPTAB

TC

+2

# IF +

TC

CCSHOLE

AD

ONE

# IF -

TS

DSMAG

INDEX

COUNT

MASK

DSMSK

EXTEND

SU

CODE

EXTEND

DFRNT

BZF

DSL

# SAME

INDEX

COUNT

CS

DSMSK

# MASK WITH 77740, 76037, 76777, OR 74037

MASK

DSMAG

AD

CODE

CS

A

INDEX

DSREL

XCH

DSPTAB

EXTEND

BZMF

DSL

# DSPTAB ENTRY WAS -

INCR

NOUT

# DSPTAB ENTRY WAS +

RELINT

DSL

TC

DSEXIT

DSMSK

OCT

37

OCT

1740

OCT

2000

OCT

3740

# FOR 11DSPIN, PUT REL ADDRESS OF DSPTAB ENTRY INTO A, 1 IN BIT11 OR 0 IN  
# BIT11 OF CODE.

11DSPIN

TS

DSREL

CAF

TWO

TS

COUNT

XCH

Q

# MUST USE SAME RETURN AS DSPIN

TS

DSEXIT

TC

DSPIN1

DSPOCTIN

TC

DSPIN

# SO DSPOCTWO DOESNT USE SWCALL

CAF

+2

TC

BANKJUMP

ENDSPOCT

CADR

OCTBACK

```
1 # DSPALARM FINDS TC NVSUBEND IN ENTRET FOR NVSUB INITIATED ROUTINES
2 # ABORT WITH 01501.
3
4 #
5 # DSPALARM FINDS TC ENDOFJOB IN ENTRET FOR KEYBOARD INITIATED ROUTINES.
6 # DC TC ENTRET.
7
8 PREDSPAL CS VD1
9 TS DSPCOUNT
10 DSPALARM CS NVSBENDL
11 AD ENTEXIT
12 EXTEND
13 BZF CHARALRM +2
14 CS MONADR # IF THIS IS A MONITOR, KILL IT
15 AD ENTEXIT
16 EXTEND
17 BZF +2
18 TC CHARALRM
19 TC KILMONON
20 TC FALTON
21 TC PASTVB # PUT MONITOR VERB BACK IN VERBREG
22 CHARALRM TC FALTON # NO NVSUB INITATED. TURN ON OPR ERROR
23 TC ENDOFJOB
24 TC POODOO
25
26 MONADR OCT 01501
27 NVSBENDL GENADR PASTVB
28 TC NVSUBEND
29
30 # ALMCYCLE TURNS ON CHECK FAIL LIGHT, REDISPLAYS THE ORIGINAL VERB THAT
31 # WAS EXECUTED, AND RECYCLES TO EXECUTE THE ORIGINAL VERB/NOUN COMBINATION
32 # THAT WAS LAST EXECUTED. USED FOR BAD DATA DURING LOAD VERBS AND BY
33 # MCTBS. ALSO BY MMCHANG IF 2 NUMERICAL CHARACTERS WERE NOT PUNCHED IN
34 # FOR MM CODE.
35
36 SETLOC MID7 +1
37 COUNT* $$/PIN
38 ALMCYCLE TC FALTON # TURN ON CHECK FAIL LIGHT.
39 CS VERBSAVE # GET ORIGINAL VERB THAT WAS EXECUTED
40 TS REQRET # SET FOR ENTPASO
41 TC BANKCALL # PUTS ORIGINAL VERB INTO VERBREG AND
42 CADR UPDATVB -1 # DISPLAYS IT IN VERB LIGHTS.
43 TC POSTJUMP
44 ENDALM CADR ENTER
45
46 # MMCHANG USES NOUN DISPLAY UNTIL ENTER. THEN IT USES MODE DISP.
47 # IT GOES TO MODROUT WITH THE NEW M M CODE IN A, BUT NOT DISPLAYED IN
48 # MM LIGHTS.
49 # IT DEMANDS 2 NUMERICAL CHARACTERS BE PUNCHED IN FOR NEW MM CODE.
50 # IF NOT, IT RECYCLES.
```



```
1 SETLOC DSP2BIT +10D
2
3
4 MMCHANG COUNT* $$/PIN
5 TC REQMM # ENTPASHI ASSUMES THE TC REQMM AT MMCHANG
6 # IF THIS MOVES AT ALL, MUST CHANGE
7 # MMADREF AT ENTPASHI.
8 CAF BIT5
9 AD DSPCOUNT # OCT20 = ND2.
10 # DSPCOUNT MUST = -ND2.
11 EXTEND
12 BZF +2
13 TC ALMCYCLE # DSPCOUNT NOT= -ND2. ALARM AND RECYCLE.
14 CAF ZERO
15 XCH NOUNREG # DSPCOUNT = -ND2.
16 TS MPAC
17 CAF ND1
18 TS DSPCOUNT
19 TC BANKCALL
20 CADR 2BLANK
21 CS VD1 # BLOCK NUM CHAR IN
22 TS DSPCOUNT
23 CA MPAC
24 TC POSTJUMP
25 CADR MODROUTB # GO THRU STANDARD LOC.
26
27 MODROUTB = V37
28 REQMM CS Q
29 TS REQRET
30 CAF ND1
31 TS DSPCOUNT
32 CAF ZERO
33 TS NOUNREG
34 TC BANKCALL
35 CADR 2BLANK
36 TC FLASHON
37 CAF ONE
38 TS DECBRNCH # SET FOR DEC
39 TC ENTEXTIT
40
41 # VBRQEXEC ENTERS REQUEST TO EXEC FOR ANY ADDRESS WITH ANY PRIORITY.
42 # IT DOES ENDOFJOB AFTER ENTERING REQUEST. DISPLAY SYST IS RELEASED.
43 # IT ASSUMES NOUN 26 HAS BEEN PRELOADED WITH
44 # COMPONENT 1 PRIORITY (BITS 10-14) BIT1=0 FOR NOVAC, BIT1=1 FOR FINDVAC.
45 # COMPONENT 2 JOB ADRES (12 BIT)
46 # COMPONENT 3 BBCON
47
48 VBRQEXEC CAF BIT1
49 MASK DSPTEM1
50 CCS A
```

```
1
2 TC SETVAC # IF BIT1 = 1, FINDVAC
3 CAF TCNOVAC # IF BIT1 = 0, NOVAC
4 REQEX1 TS MPAC # TC NOVAC OR TC FINDVAC INTO MPAC
5 CS BIT1
6 MASK DSPTM1
7 REQUESTC TS MPAC +4 # PRIO INTO MPAC+4 AS A TEMP
8 TC RELDSP
9 CA ENDINST
10 TS MPAC +3 # TC ENDOFJOB INTO MPAC+3
11 EXTEND
12 DCA DSPTM1 +1 # JOB ADRES INTO MPAC+1
13 DXCH MPAC +1 # BBCON INTO MPAC+2
14 CA MPAC +4 # PRIO IN A
15 INHINT
16 TC MPAC
17
18 SETVAC CAF TCFINDVC
19 TC REQEX1
20
21 # VBRQWAIT ENTERS REQUEST TO WAITLIST FOR ANY ADDRESS WITH ANY DELAY.
22 # IT DOES ENDOFJOB AFTER ENTERING REQUEST. DISPLAY SYST IS RELEASED.
23 # IT ASSUMES NOUN 26 HAS BEEN PRELOADED WTIH
24 # COMPONENT 1 DELAY (LOW BITS)
25 # COMPONENT 2 TASK ADRES (12 BIT)
26 # COMPONENT 3 BBCON
27
28 VBRQWAIT CAF TCWAIT
29 TS MPAC # TC WAITLIST INTO MPAC
30 CA DSPTM1 # TIME DELAY
31 ENDRQWT TC REQUESTC -1
32
33 # REQUESTC WILL PUT TASK ADRES INTO MPAC+1, BBCON INTO MPAC+2,
34 # TC ENDOFJOB INTO MPAC+3. IT WILL TAKE TIME DELAY OUT OF MPAC+4 AND
35 # LEAVE IT IN A, INHINT AND TC MPAC.
36
37 VBPROC SETLOC NVSBENDL +1
38 COUNT* $$/PIN
39 CAF ONE # PROCEED WITHOUT DATA
40 TS LOADSTAT
41 TC KILMONON # TURN ON KILL MONITOR BIT
42 TC RELDSP
43 TC FLASHOFF
44 TC RECALST # SEE IF THERE IS ANY RECALL FROM ENDIDLE
45
46 VBTERM CS ONE
47 TC VBPROC +1 # TERM VERB SETS LOADSTAT NEG
48
49
50
51
52
53
54
55
56
57
58
59
60
```



# PINBALL\_GAME\_BUTTONS\_AND\_LIGHTS

# PROCKEY PERFORMS THE SAME FUNCTION AS VBPROC. IT MUST BE CALLED UNDER  
# EXECUTIVE CONTROL, WITH CHRPRIO.

|         |     |          |                                            |
|---------|-----|----------|--------------------------------------------|
| PROCKEY | CAF | ZERO     | # SET REQRET FOR ENTER PASS 0.             |
|         | TS  | REQRET   |                                            |
|         | CS  | VD1      | # BLOCK NUMERICAL CHARACTERS, SIGNS, CLEAR |
|         | TS  | DSPCOUNT |                                            |
|         | TC  | VBPROC   |                                            |

# VBRESEQ WAKES ENDIDLE AT SAME LINE AS FINAL ENTER OF LOAD (L+3).  
# (MAIN USE IS INTENDED AS RESPONSE TO INTERNALLY INITIATED FLASHING  
# DISPLAYS IN ENDIDLE. SHOULD NOT BE USED WITH LOAD VERBS, PLEASE PERFORM,  
# OR PLEASE MARK VERBS BECAUSE THEY ALREADY USE L+3 IN ANOTHER CONTEXT.)

|         |    |           |                              |
|---------|----|-----------|------------------------------|
| VBRESEQ | CS | ZERO      | # MAKE IT LOOK LIKE DATA IN. |
|         | TC | VBPROC +1 |                              |

# FLASH IS TURNED OFF BY PROCEED WITHOUT DATA, TERMINATE, RESEQUENCE,  
# END OF LOAD.

## # KEY RELEASE ROUTINE

#  
# THIS ROUTINE ALWAYS TURNS OFF THE UPACT LIGHT AND ALWAYS CLEARS DSPLOCK.

#  
# THE HIGHEST PRIORITY FUNCTION OF THE KEY RELEASE BUTTON IS THE  
# UNSUSPENDING OF A SUSPENDED MONITOR WHICH WAS EXTERNALLY INITIATED.  
# THIS FUNCTION IS ACCOMPLISHED BY CLEARING DSPLOCK AND TURNING OFF  
# THE KEY RELEASE LIGHT IF BOTH DSPLIST AND CADRSTOR ARE EMPTY.

#  
# IF NO SUCH MONITOR EXISTS, THEN RELDSP IS EXECUTED TO CLEAR DSPLOCK  
# AND THE EXTERNAL MONITOR BIT (FREEING THE DISPLAY SYSTEM FOR INTERNAL  
# USE), TURN OFF THE KEY RELEASE LIGHT, AND WAKE UP ANY JOB IN DSPLIST.

#  
# IN ADDITION IF THERE IS A JOB IN ENDIDLE, THEN CONTROL IS TRANSFERRED  
# TO PINBRNCH (IN DISPLAY INTERFACE ROUTINE) TO RE-EXECUTE THE SERIES OF  
# NVSUB CALLS ETC. THAT PRECEDED THE ENDIDLE CALL STILL AWAITING RESPONSE.

# THIS FEATURE IS INTENDED FOR USE WHEN THE OPERATOR HAS BEEN REQUESTED TO  
# RESPOND TO SOME INTERNAL ACTION THAT USED ENDIDLE, BUT HE HAS WRITTEN  
# OVER THE INFORMATION ON THE DISPLAY PANEL BY SOME DISPLAYS OF HIS OWN

# INITIATION WHICH DO NOT SERVE AS RESPONSES. HITTING KEYRLSE WILL  
# RE-ESTABLISH THE DISPLAYS TO THE STATE THEY WERE IN BEFORE HE OBSCURED  
# THEM, SO THAT HE CAN SEE THE WAITING REQUEST. THIS WORKS ONLY FOR

# INTERNAL PROGRAMS THAT USED ENDIDLE THROUGH MARGARETS DISPLAY  
# SUBROUTINES.

|          |        |          |                                          |
|----------|--------|----------|------------------------------------------|
| VBRELDSP | CS     | BIT3     |                                          |
|          | EXTEND |          |                                          |
|          | WAND   | DSALMOUT | # TURN OFF UPACT LITE                    |
|          | CCS    | 21/22REG | # OLD DSPLOCK                            |
|          | CAF    | BIT14    |                                          |
|          | MASK   | MONSAVE1 | # EXTERNAL MONITOR BIT (EMB)             |
|          | CCS    | A        |                                          |
|          | TC     | UNSUSPEN | # OLD DSPLOCK AND EMB BOTH 1, UNSUSPEND. |
| TSTLTS4  | TC     | RELDSP   | # NOT UNSUSPENDING EXTERNAL MONITOR,     |
|          | CCS    | CADRSTOR | # RELEASE DISPLAY SYSTEM AND             |
|          | TC     | +2       | # DO RE-ESTABLISH IF CADRSTOR IS FULL.   |
|          | TC     | ENDOFJOB |                                          |
|          | TC     | POSTJUMP |                                          |
|          | CADR   | PINBRNCH |                                          |
| UNSUSPEN | CAF    | ZERO     | # EXTERNAL MONITOR IS SUSPENDED,         |
|          | TS     | DSPLOCK  | # JUST UNSUSPEND IT BY CLEARING DSPLOCK. |
|          | CCS    | CADRSTOR | # TURN KEY RELEASE LIGHT OFF IF BOTH     |
|          | TC     | ENDOFJOB | # CADRSTOR AND DSPLIST ARE EMPTY.        |
|          | TC     | RELDSP1  |                                          |
|          | TC     | ENDOFJOB |                                          |

ENDRELDS      EQUALS

```
1 # NVSUB IS USED FOR SUBROUTINE CALLS FROM WITHIN COMPUTER. IT CAN BE
2 # USED TO CALL THE COMBINATION OF ANY DISPLAY, LOAD, OR MONITOR VERB
3 # TOGETHER WITH ANY NOUN AVAILABLE TO THE KEYBOARD.
4 # PLACE OVVVVVVVNNNNNNN INTO A.
5 # V'S ARE THE 7-BIT VERB CODE. N'S ARE THE 7-BIT NOUN CODE.
6 #
7 # IF NVSUB IS CALLED WITH THE FOLLOWING NEGATIVE NUMBERS (RATHER THAN THE
8 # VERB-NOUN CODE) IN A, THEN THE DISPLAY IS BLANKED AS FOLLOWS -
9 # -4 FULL BLANK, -3 LEAVE MODE, -2 LEAVE MODE AND VERB, -1 BLANK R'S ONLY.
10 #
11 # NVSUB CAN BE USED WITH MACH CADR TO BE SPEC BY PLACING THE CADR INTO
12 # MPAC+2 BEFORE THE STANDARD NVSUB CALL.
13 #
14 # NVSUB RETURNS TO 2+ CALLING LOC AFTER PERFORMING TASK, IF DISPLAY
15 # SYSTEM IS AVAILABLE. THE NEW NOUN AND VERB CODES ARE DISPLAYED.
16 # IF V'S =0, THE NEW NOUN CODE IS DISPLAYED ONLY (RETURN WITH NO FURTHER
17 # ACTION). IF N'S =0, THE NEW VERB CODE IS DISPLAYED ONLY (RETURN WITH NO
18 # FURTHER ACTION).
19 #
20 # IT RETURNS TO 1+ CALLING LOC WITHOUT PERFORMING TASK, IF DISPLAY
21 # SYSTEM IS BLOCKED (NOTHING IS DISPLAYED IN THIS CASE).
22 # IT DOES TC ABORT (WITH OCT 01501) IF IT ENCOUNTERS A DISPLAY PROGRAM
23 # ALARM CONDITION BEFORE RETURN TO CALLER.
24 #
25 # THE DISPLAY SYSTEM IS BLOCKED BY THE DEPRESSION OF ANY
26 # KEY, EXCEPT ERROR LIGHT RESET.
27 #
28 # IT IS RELEASED BY THE KEY RELEASE BUTTON, ALL EXTENDED VERBS,
29 # PROCED WITHOUT DATA, TERMINATE, RESEQUENCE, INITIALIZE EXECUTIVE,
30 # RECALL PART OF RECALTST IF ENDIDLE WAS USED,
31 # VB = REQUEST EXECUTIVE, VB = REQUEST WAITLIST,
32 # MONITOR SET UP.
33 #
34 # THE DISPLAY SYSTEM IS ALSO BLOCKED BY THE EXTERNAL MONITOR BIT, WHICH
35 # INDICATES AND EXTERNALLY INITIATED MONITOR IS RUNNING (SEE MONITOR).
36 #
37 # A NVSUB CALL THAT PASSES DSPLOCK AND THE EXTERNAL MONITOR BIT ENDS OLD
38 # MONITOR.
39 #
40 # DSPLOCK IS THE INTERLOCK FOR USE OF KEYBOARD AND DISPLAY SYSTEM WHICH
41 # LOCKS OUT INTERNAL USE WHENEVER THERE IS EXTERNAL KEYBOARD ACTION.
42 #
43 # NVSUB SHOULD BE USED TWICE IN SUCCESSION FOR 'PLEASE PERFORM' SITUATIONS
44 # (SIMILARLY FOR PLEASE MARK). FIRST PLACE THE CODED NUMBER FOR WHAT
45 # ACTION IS DESIRED OF OPERATOR INTO THE REGISTERS REFERRED TO BY THE
46 # 'CHECKLIST' NOUN. GO TO NVSUB WITH A DISPLAY VERB AND THE 'CHECKLIST'
47 # NOUN. GO TO NVSUB AGAIN WITH THE 'PLEASE PERFORM' VERB AND ZEROS IN THE
48 # LOW 7 BITS. THIS 'PASTES UP' THE 'PLEASE PERFORM' VERB INTO THE VERB
49 # LIGHTS.
50 #
51 # NVMONOPT IS AN ENTRY SIMILAR TO NVSUB, BUT REQUIRING AN ADDITIONAL
```

```
1 # PARAMETER IN L. IT SHOULD BE USED ONLY WITH A MONITOR VERB-NOUN CODE IN
2 # A. AFTER EACH MONITOR DISPLAY A *PLEASE* VERB WILL BE PASED INT THE VERB
3 # LIGHTS OR DATA WILL BE BLANKED (OR BOTH) ACCORDING TO THE OPTIONS
4 # SPECIFIED IN L. IF BITS 8-14 OF L ARE OTHER THAN ZERO, THEN THEY WILL
5 # BE INTERPRETED AS A VERB CODE AND PASTED IN THE VERB LIGHTS. (THIS VERB
6 # CODE SHOULD DESIGNATE ONE OF THE *PLEASE* VERBS.) IF BITS 1-3 OF L ARE
7 # OTHER THAN ZERO, THEN THEY WILL BE USED TO BLANK DATA BY BEING FED TO
8 # BLANKSUB. IF NVMONOPT IS USED WITH A VERB OTHER THAN A MONITOR VERB,
9 # THE PARAMETER IN L HAS NO EFFECT.
10 #
11 # NVSUB IN FIXED-FIXED PLACES 2+CALLING LOC INTO NVQTEM, TC NVSUBEND INTO
12 # ENTRET. (THIS WILL RESTORE OLD CALLING BANK BITS)
```

```
13 SETLOC ENDALM +1
```

```
14
15
16
17 COUNT* $$/PIN
18 NVSUB LXCH 7 # ZERO NVMONOPT OPTIONS
19 NVMONOPT TS NVTEMP
20 CAF BIT14
21 MASK MONSAVE1 # EXTERNAL MONITOR BIT
22 AD DSPLOCK
23 CCS A
24 TC Q # DSP SYST BLOCKED, RET TO 1+ CALLING LOC
25 NVSBCOM CAF ONE # DSP SYST AVAILABLE.
26 AD Q
27 TS NVQTEM # 2+ CALLING LOC INTO NVQTEM
28 LXCH MONSAVE2 # STORE NVMONOPT OPTIONS
29 TC KILMONON # TURN ON KILL MONITOR BIT
30 NVSUBCOM CAF NVSBBBNK
31
32 XCH BBANK
33 EXTEND # SAVE OLD SUPERBITS
34 ROR SUPERBNK
35 TS NVBNKTEM
36 CAF PINSUPBT
37 EXTEND
38 WRITE SUPERBNK
39 TC NVSUBB # GO TO NVSUB1 THRU STANDARD LOC
40 NVSBBBNK EBANK= DSPCOUNT
41 BBCON NVSUB1
42
43 PINSUPBT = NVSBBBNK # CONTAINS THE PINBALL SUPERBITS.
44 NVSUBEND DXCH NVQTEM # NVBNKTEM MUST = NVQTEM+1
45 TC SUPDXCHZ # DTCB WITH SUPERBIT SWITCHING
```

```
46 SETLOC ENDRQWT +1
```

```
47 COUNT* $$/PIN
```

```
48 # BLANKDSP BLANKS DISPLAY ACCORDING TO OPTION NUMBER IN NVTEMP AS FOLLOWS
49
50
51
52
53
54
55
56
57
58
59
60
```

# -4 FULL BLANK, -3 LEAVE MODE, -2 LEAVE MODE AND VERB, -1 BLANK R'S ONLY.

BLANKDSP AD SEVEN # 7,8,9, OR 10 (A HAD 0,1,2,OR 3)

INHINT

TS

CODE

# BLANK SPECIFIED DSPTABS

CS

BIT12

INDEX

CODE

XCH

DSPTAB

CCS

A

INCR

NOUT

TC

+1

CCS

CODE

TC

BLANKDSP +2

RELINT

INDEX

NVTEMP

TC

+5

TC

+1

# NVTEMP HAS -4 (NEVER TOUCH MODREG)

TS

VERBREG

#

-3

TS

NOUNREG

#

-2

TS

CLPASS

#

-1

CS

VD1

TS

DSPCOUNT

TC

FLASHOFF

# PROTECT AGAINST INVISIBLE FLASH

TC

ENTSET -2

# ZEROS REQRET

NVSUB1

CAF

ENTSET

# IN BANK

TS

ENTRET

# SET RETURN TO NVSUBEND

CCS

NVTEMP

# WHAT NOW

TC

+4

# NORMAL NVSUB CALL (EXECUTE VN OR PASTE)

TC

GODSPALM

TC

BLANKDSP

# BLANK DISPLAY AS SPECIFIED

TC

GODSPALM

CAF

LOW7

MASK

NVTEMP

TS

MPAC +3

# TEMP FOR NOUN (CANT USE MPAC. DSPDECVN

CA

NVTEMP

# USES MPAC, +1, +2).

TS

EDOP

# RIGHT 7

CA

EDOP

TS

MPAC +4

# TEMP FOR VERB (CANT USE MPAC+1. DSPDECVN

# USES MPAC, +1, +2).

CCS

MPAC +3

# TEST NOUN

TC

NVSUB2

# IF NOUN NOT +0, GO ON

CA

MPAC +4

TC

UPDATVB -1

# IF NOUN = +0, DISPLAY VERB, THEN RETURN

CAF

ZERO

# XERO REQRET SO THAT PASTED VERBS CAN

TS

REQRET

# BE EXECUTED BY OPERATOR.

TC

NVSUBEND

ENTSET

NVSUB2

CCS

MPAC +4

# TEST VERB

TC

+4

# IF VERB NOT +0, GO ON

CA

MPAC +3



```
1
2 TC UPDATNN -1 # IF VERB = +0, DISPLAY NOUN, THEN RETURN
3 TC NVSUBEND
4 CA MPAC +2 # TEMP FOR MACH CADR TO BE SPEC. (DSPDECVN
5 TS MPAC +5 # USES MPAC, +1, +2)
6 CA MPAC +4
7 TC UPDATVB -1 # IF BOTH NOUN AND VERB NOT +0, DISPLAY
8 CA MPAC +3 # BOTH AND GO TO ENTPAS0.
9 TC UPDATNN -1
10 CAF ZERO
11 TS LOADSTAT # SET FOR WAITING FOR DATA CONDITION
12 TS CLPASS
13 TS REQRET # SET REQRET FOR PASS 0.
14 CA MPAC +5 # RESTORES MACH CADR TO BE SPEC TO MPAC+2
15 TS MPAC +2 # FOR USE IN INTMCTBS (IN ENTPAS0).
16 ENDNVSB1 TC ENTPAS0
17
18 # IF INTERNAL MACH CADR TO BE SPECIFIED, MPAC+2 WILL BE PLACED INTO
19 # NOUNCADR IN ENTPAS0 (INTMCTBS).
20
21 SETLOC NVSUBEND +2
22 COUNT* $$/PIN
23
24 KILMONON CAF BIT15 # FORCE BIT 15 OF MONSAVE1 TO 1.
25 TS MONSAVE1 # THIS IS THE KILL MONITOR BIT.
26 TS MONSAVE1 # TURN OFF BIT 14, THE EXTERNAL
27 TC Q # MONITOR BIT.
28
29 # LOADSTAT +0 INACTIVE (WAITING FOR DATA). SET BY NVSUB
30 # +1 PROCEED NO DATA. SET BY SPECIAL VERB
31 # -1 TERMINATE. SET BY SPECIAL VERB.
32 # -0 DATA IN SET BY END OF LOAD ROUTINE
33 # OR RESEQUENCE SET BY VERB 32
34 #
35 # L TO ENDIDLE (FIXED FIXED)
36 # ROUTINES THAT REQUEST LOADS THROUGH NVSUB SHOULD USE ENDIDLE WHILE
37 # WAITING FOR THE DATA TO BE LOADED. ENDIDLE PUTS CURRENT JOB TO SLEEP.
38 # ENDIDLE CANNOT BE CALLED FROM ERASABLE OR F/F MEMORY,
39 # SINCE JOB SLEEP AND JOBWAKE CAN HANDLE ONLY FIXED BANKS.
40 # RECALST TESTS LOADSTAT AND WAKES JOB UP TO,
41 # L+1 FOR TERMINATE
42 # L+2 FOR PROCEED WITHOUT DATA
43 # L+3 FOR DATA IN, OR RESEQUENCE
44 # IT DOES NOTHING IF LOADSTAT INDICATES WAITING FOR DTA.
45 #
46 # ENDIDLE ABORTS (WITH CODE 1206) IF A SECOND JOB ATTEMPTS TO GO TO SLEEP
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```



```
1 # IN PINBALL. IN PARTICULAR, IF AN ATTEMPT IS MADE TO GO TO ENDIDLE WHEN
2 # 1) CADRSTOR NOT= +0. THIS IS THE CASE WHERE THE CAPACITY OF ENDIDLE IS
3 # EXCEEDED. (+-NZ INDICATES A JOB IS ALREADY ASLEEP DUE TO ENDIDLE.)
4 # 2) DSPLIST NOT= +0. THIS INDICATES A JOB IS ALREADY ASLEEP DUE TO
5 # NVSUBUSY.
6
7 ENDIDLE LXCH Q # RETURN ADDRESS INTO L.
8 TC ISCADR+0 # ABORT IF CADRSTOR NOT= +0
9 TC ISLIST+0 # ABORT IF DSPLIST NOT= +0
10 CA L # DONT SET DSPLOC TO 1 SO CAN USE
11 MASK LOW10 # ENDIDLE WITH NVSUB INITIATED MONITOR.
12 AD FBANK # SAME STRATEGY FOR CADR AS MAKECADR.
13 TS CADRSTOR
14 TC JOBSLEEP
15
16 ENDINST TC ENDOFJOB
17
18 ISCADR+0 CCS CADRSTOR # ABORTS (CODE 01206) IF CADRSTOR NOT= +0.
19 TC DSPABORT # RETURNS IF CADRSTOR = +0.
20 TC Q
21 TC DSPABORT
22
23 ISLIST+0 CCS DSPLIST # ABORTS (CODE 01206) IF DSPLIST NOT= +0.
24 TC DSPABORT # RETURNS IF DSPLIST = +0.
25 TC Q
26 DSPABORT TC POOD00
27 OCT 01206
28
29 # JAMTERM ALLOWS PROGRAMS TO PERFORM THE TERMINATE FUNCTION.
30 # IT DOES ENDOFJOB.
31
32 JAMTERM CAF PINSUPBT
33 EXTEND
34 WRITE SUPERBNK
35 CAF 34DEC
36 TS REQRET # LEAVE ENTER SET FOR ENTPASS0.
37 CS VD1
38 TS DSPCOUNT
39 TC POSTJUMP
40 CADR VBTERM
41
42 34DEC DEC 34
43
44 # JAMPROC ALLOWS PROGRAMS TO PERFORM THE PROCEED/PROCEED WITHOUT DATA
45 # FUNCTION. IT DOES ENDOFJOB.
```

```
1 JAMPROC CAF PINSUPBT
2 EXTEND
3
4 WRITE SUPERBNK
5 CAF 33DEC
6 TS REQRET # LEAVE ENTER SET FOR ENTPASS0.
7
8 CS VD1
9 TS DSPCOUNT
10 TC POSTJUMP
11
12 CADR VBPROC
13
14 33DEC DEC 33
15
16 # BLANKSUB BLANKS ANY COMBINATION OF R1, R2, R3.
17 # CALL WITH BLANKING CODE IN A.
18 # BIT1=1 BLANKS R1, BIT2=1 BLANKS R2, BIT3=1 BLANKS R3.
19 # ANY COMBINATION OF THESE BITS IS ACCEPTED.
20 #
21 # DSPCOUNT IS RESTORED TO STATE IT WAS IN BEFORE BLANKSUB WAS EXECUTED.
22
23 BLANKSUB MASK SEVEN
24 TS NVTEMP # STORE BLANKING CODE IN NVTEMP.
25 CAF BIT14
26 MASK MONSAVE1 # EXTERNAL MONITOR BIT
27
28 AD DSPLOCK
29 CCS A
30 TC Q # DSP SYST BLOCKED. RET TO 1+ CALLING LOC
31 INCR Q # DSP SYST AVAILABLE
32 # SET RETURN FOR 2+ CALLING LOC
33
34 CCS NVTEMP
35 TCF +2
36 TC Q # NOTHING TO BLANK. RET TO 2+ CALLING LOC
37 LXCH Q # SET RETURN FOR 2 + CALLING LOC
38
39 CAF BLNKBBNK
40 XCH BBANK
41 EXTEND
42 ROR SUPERBNK # SAVE OLD SUPERBITS.
43 DXCH BUF
44 CAF PINSUPBT
45
46 EXTEND
47 WRITE SUPERBNK
48 TC BLNKSUB1
49
50 BLNKBBNK EBANK= DSPCOUNT
51 ENDBLFF BBCON BLNKSUB1
52
53 SETLOC ENDRELDS
54
55 BLNKSUB1 COUNT* $$/PIN
56 CA DSPCOUNT # SAVE OLD DSPCOUNT FOR LATER RESTORATION
```

```
1
2 TS BUF +2
3 CAF BIT1 # TEST BIT1. SEE IF R1 TO BE BLANKED.
4 TC TESTBIT
5 CAF R1D1
6 TC 5BLANK -1
7 CAF BIT2 # TEST BIT2. SEE IF R2 TO BE BLANKED.
8 TC TESTBIT
9 CAF R2D1
10 TC 5BLANK -1
11 CAF BIT3 # TEST BIT3. SEE IF R3 TO BE BLANKED.
12 TC TESTBIT
13 CAF R3D1
14 TC 5BLANK -1
15 CA BUF +2 # RESTORE DSPCOUNT TO STATE IT HAD
16 TS DSPCOUNT # BEFORE BLANKSUB.
17 DXCH BUF # CALL L+2 DIRECTLY.
18 TC SUPDXCHZ +1 # DTCB WITH SUPERBIT SWITCHING
19
20 TESTBIT MASK NVTEMP # NVTEMP CONTAINS BLANKING CODE.
21 CCS A
22 TC Q # IF CURRENT BIT = 1, RETURN TO L+1.
23 INDEX Q # IF CURRENT BIT = 0, RETURN TO L+3.
24 TC 2
25
26 ENDBSUB1 EQUALS
27
28 # DSPMM DOES NOT DISPLAY MODREG DIRECTLY. IT PUTS IN EXEC REQUEST WITH
29 # PRIO 30000 FOR DSPMMJB AND RETURNS TO CALLER.
30 #
31 # IF MODREG CONTAINS -0, DSPMMJB BLANKS THE MODE LIGHTS.
32 #
33 # DSPMM MUST BE IN BANK 27 OR LOWER, SO IT CAN BE CALLED VIA BANKCALL.
34
35 BANK 7
36 SETLOC PINBALL4
37 BANK
38
39 COUNT* $$/PIN
40 DSPMM XCH Q
41 TS MPAC
42 INHINT
43 CAF CHRPRIO
44 TC NOVAC
45 EBANK= DSPCOUNT
46 2CADR DSPMMJB
47
48 RELINT
49 ENDSPMM TC MPAC
50
51
52
53
54
55
56
57
58
59
60
```

# DSPMM PLACE MAJOR MODE CODE INTO MODREG

SETLOC ENDBSUB1

COUNT\* \$\$/PIN

DSPMMJB

CAF

MD1

# GETS HERE THRU DSPMM

XCH

DSPCOUNT

TS

DSPMMTEM

# SAVE DSPCOUNT

CCS

MODREG

AD

ONE

TC

DSPDECVN

# IF MODREG IS + OR +0, DISPLAY MODREG

TC

+2

# IF MODREG IS -NZ, DO NOTHING

TC

2BLANK

# IF MODREG IS -0, BLANK MM

XCH

DSPMMTEM

# RESTORE DSPCOUNT

TS

DSPCOUNT

TC

ENDOFJOB

# RECALTST IS ENTERED DIRECTLY AFTER DATA IS LOADED (OR RESEQUENCE VERB IS  
# EXECUTED), TERMINATE VERB IS EXECUTED, OR PROCEED WITHOUT DATA VERB IS  
# EXECUTED. IT WAKES UP JOB THAT DID TC ENDIDLE.#  
# IF CADDRSTOR NOT= +0, IT PUTS +0 INTO DSPLOCK, AND TURNS OFF KEY RLSE  
# LIGHT IF DSPLIST IS EMPTY (LEAVES KEY RLSE LIGHT ALONE IF NOT EMPTY).

RECALTST

CCS

CADDRSTOR

TC

RECAL1

RECAL1

TC

ENDOFJOB

# NORMAL EXIT IF KEYBOARD INITIATED

CAF

ZERO

XCH

CADDRSTOR

INHINT

TC

JOBWAKE

CCS

LOADSTAT

TC

DOPROC

# + PROCEED WITHOUT DATA

TC

ENDOFJOB

# PATHOLOGICAL CASE EXIT

TC

DOTERM

# - TERMINATE

RECAL2

CAF

TWO

# -0 DATA IN OR RESEQUENCE

INDEX

LOCCTR

AD

LOC

# LOC IS + FOR BASIC JOBS

INDEX

LOCCTR

TS

LOC

CA

NOUNREG

# SAVE VERB IN MPAC, NOUN IN MPAC+1 AT

TS

L

# TIME OF RESPONSE TO ENDIDLE FOR

CA

VERBREG

# POSSIBLE LATER TESTING BY JOB THAT HAS

INDEX

LOCCTR

# BEEN WAKED UP.

DXCH

MPAC

RELINT

RECAL3

TC

RELDSP

TC

ENDOFJOB



|    |        |     |        |    |
|----|--------|-----|--------|----|
| 1  |        |     |        | 1  |
| 2  | DOTERM | CAF | ZERO   | 2  |
| 3  |        | TC  | RECAL2 | 3  |
| 4  |        |     |        | 4  |
| 5  | DOPROC | CAF | ONE    | 5  |
| 6  |        | TC  | RECAL2 | 6  |
| 7  |        |     |        | 7  |
| 8  |        |     |        | 8  |
| 9  |        |     |        | 9  |
| 10 |        |     |        | 10 |
| 11 |        |     |        | 11 |
| 12 |        |     |        | 12 |
| 13 |        |     |        | 13 |
| 14 |        |     |        | 14 |
| 15 |        |     |        | 15 |
| 16 |        |     |        | 16 |
| 17 |        |     |        | 17 |
| 18 |        |     |        | 18 |
| 19 |        |     |        | 19 |
| 20 |        |     |        | 20 |
| 21 |        |     |        | 21 |
| 22 |        |     |        | 22 |
| 23 |        |     |        | 23 |
| 24 |        |     |        | 24 |
| 25 |        |     |        | 25 |
| 26 |        |     |        | 26 |
| 27 |        |     |        | 27 |
| 28 |        |     |        | 28 |
| 29 |        |     |        | 29 |
| 30 |        |     |        | 30 |
| 31 |        |     |        | 31 |
| 32 |        |     |        | 32 |
| 33 |        |     |        | 33 |
| 34 |        |     |        | 34 |
| 35 |        |     |        | 35 |
| 36 |        |     |        | 36 |
| 37 |        |     |        | 37 |
| 38 |        |     |        | 38 |
| 39 |        |     |        | 39 |
| 40 |        |     |        | 40 |
| 41 |        |     |        | 41 |
| 42 |        |     |        | 42 |
| 43 |        |     |        | 43 |
| 44 |        |     |        | 44 |
| 45 |        |     |        | 45 |
| 46 |        |     |        | 46 |
| 47 |        |     |        | 47 |
| 48 |        |     |        | 48 |
| 49 |        |     |        | 49 |
| 50 |        |     |        | 50 |
| 51 |        |     |        | 51 |
| 52 |        |     |        | 52 |
| 53 |        |     |        | 53 |
| 54 |        |     |        | 54 |
| 55 |        |     |        | 55 |
| 56 |        |     |        | 56 |
| 57 |        |     |        | 57 |
| 58 |        |     |        | 58 |
| 59 |        |     |        | 59 |
| 60 |        |     |        | 60 |

## # MISCELLANEOUS SERVICE ROUTINES IN FIXED/FIXED

SETLOC ENDBLFF

COUNT\* \$\$/PIN

# SETNCADR E CADR ARRIVES IN A. IT IS STORED IN NOUNCADR. EBANK BITS  
# ARE SET. E ADRES IS DERIVED AND PUT INTO NOUNADD.

SETNCADR TS NOUNCADR # STORE ECADR  
TS EBANK # SET EBANK BITS  
MASK LOW8  
AD OCT1400  
TS NOUNADD # PUT E ADRES INTO NOUNADD  
TC Q

# SETNADD GETS E CADR FROM NOUNCADR, SETS EBANK BITS, DERIVES  
# E ADRES AND PUTS IT INTO NOUNADD.

SETNADD CA NOUNCADR  
TCF SETNCADR +1

# SETEBANK E CADR ARRIVES IN A. EBANK BITS ARE SET. E ADRES IS  
# DERIVED AND LEFT IN A.

SETEBANK TS EBANK # SET EBANK BITS  
MASK LOW8  
AD OCT1400 # E ADRES LEFT IN A  
TC Q

R1D1 OCT 16 # THESE 3 CONSTANTS FORM A PACKED TABLE.  
R2D1 OCT 11 # DONT SEPARATE.  
R3D1 OCT 4

RIGHT5 TS CYR  
CS CYR  
CS CYR  
CS CYR  
CS CYR  
XCH CYR  
TC Q

LEFT5 TS CYL  
CS CYL  
CS CYL  
CS CYL  
CS CYL

|    |          |        |          |                                          |    |
|----|----------|--------|----------|------------------------------------------|----|
| 1  |          |        |          |                                          | 1  |
| 2  |          | XCH    | CYL      |                                          | 2  |
| 3  |          | TC     | Q        |                                          | 3  |
| 4  |          |        |          |                                          | 4  |
| 5  | SLEFT5   | DOUBLE |          |                                          | 5  |
| 6  |          | DOUBLE |          |                                          | 6  |
| 7  |          | DOUBLE |          |                                          | 7  |
| 8  |          | DOUBLE |          |                                          | 8  |
| 9  |          | DOUBLE |          |                                          | 9  |
| 10 |          | TC     | Q        |                                          | 10 |
| 11 |          |        |          |                                          | 11 |
| 12 | LOW5     | OCT    | 37       | # THESE 3 CONSTANTS FORM A PACKED TABLE. | 12 |
| 13 | MID5     | OCT    | 1740     | # DONT SEPARATE.                         | 13 |
| 14 | HI5      | OCT    | 76000    | # MUST STAY HERE                         | 14 |
| 15 |          |        |          |                                          | 15 |
| 16 | TCNOVAC  | TC     | NOVAC    |                                          | 16 |
| 17 | TCWAIT   | TC     | WAITLIST |                                          | 17 |
| 18 | TCTSKOVR | TC     | TASKOVER |                                          | 18 |
| 19 | TCFINDVC | TC     | FINDVAC  |                                          | 19 |
| 20 |          |        |          |                                          | 20 |
| 21 | CHRPRI0  | OCT    | 30000    | # EXEC PRIORITY OF CHARIN                | 21 |
| 22 |          |        |          |                                          | 22 |
| 23 | LOW11    | OCT    | 3777     |                                          | 23 |
| 24 | B12-1    | EQUALS | LOW11    |                                          | 24 |
| 25 | LOW8     | OCT    | 377      |                                          | 25 |
| 26 |          |        |          |                                          | 26 |
| 27 | VD1      | OCT    | 23       | # THESE 3 CONSTANTS FORM A PACKED TABLE. | 27 |
| 28 | ND1      | OCT    | 21       | # DONT SEPARATE.                         | 28 |
| 29 | MD1      | OCT    | 25       |                                          | 29 |
| 30 |          |        |          |                                          | 30 |
| 31 | BINCON   | DEC    | 10       |                                          | 31 |
| 32 |          |        |          |                                          | 32 |
| 33 | FALTON   | CA     | BIT7     | # TURN ON OPERATOR ERROR LIGHT.          | 33 |
| 34 |          | EXTEND |          |                                          | 34 |
| 35 |          | WOR    | DSALMOUT | # BIT 7 OF CHANNEL 11                    | 35 |
| 36 |          | TC     | Q        |                                          | 36 |
| 37 |          |        |          |                                          | 37 |
| 38 | FALTOF   | CS     | BIT7     | # TURN OFF OPERATOR ERROR LIGHT          | 38 |
| 39 |          | EXTEND |          |                                          | 39 |
| 40 |          | WAND   | DSALMOUT | # BIT 7 OF CHANNEL 11                    | 40 |
| 41 |          | TC     | Q        |                                          | 41 |
| 42 |          |        |          |                                          | 42 |
| 43 | RELDSPON | CAF    | BIT5     | # TURN ON KEY RELEASE LIGHT              | 43 |
| 44 |          | EXTEND |          |                                          | 44 |
| 45 |          | WOR    | DSALMOUT | # BIT 5 OF CHANNEL 11                    | 45 |
| 46 |          | TC     | Q        |                                          | 46 |
| 47 |          |        |          |                                          | 47 |
| 48 |          |        |          |                                          | 48 |
| 49 |          |        |          |                                          | 49 |
| 50 |          |        |          |                                          | 50 |
| 51 |          |        |          |                                          | 51 |
| 52 |          |        |          |                                          | 52 |
| 53 |          |        |          |                                          | 53 |
| 54 |          |        |          |                                          | 54 |
| 55 |          |        |          |                                          | 55 |
| 56 |          |        |          |                                          | 56 |
| 57 |          |        |          |                                          | 57 |
| 58 |          |        |          |                                          | 58 |
| 59 |          |        |          |                                          | 59 |
| 60 |          |        |          |                                          | 60 |

```
1 LODSAMPT EXTEND
2 DCA TIME2
3 DXCH SAMPTIME
4 TC Q
5
6
7 TPSL1 EXTEND # SHIFTS MPAC, +1, +2 LEFT 1
8 DCA MPAC +1 # LEAVES OVFIN D SET TO +/- 1 FOR OF/UF
9 DAS MPAC +1
10 AD MPAC
11 ADS MPAC
12 TS 7 # TS A DOES NOT CHANGE A ON OF/UF.
13 TC Q # NO NET OF/UF
14 TS MPAC+6 # MPAC +6 SET TO +/- 1 FOR OF/UF
15 TC Q
16
17 # IF MPAC, +1 ARE EACH +NZ OR +0 AND C(A)=-0, SHORTMP WRONGLY GIVES +0.
18 # IF MPAC, +1 ARE EACH -NZ OR -0 AND C(A)=+0, SHORTMP WRONGLY GIVES +0.
19 # PRSHRTMP FIXES FORST CASE ONLY, BY MERELY TESTING C(A) AND IF IT = -0,
20 # SETTING RESULT TO -0.
21 # (DO NOT USE PRSHRTMP UNLESS MPAC, +1 ARE EACH +NZ OR +0, AS THEY ARE
22 # WHEN THEY CONTAIN THE SF CONSTANTS.)
23
24 PRSHRTMP TS MPTMP
25 CCS A
26 CA MPTMP # C(A) +, DO REGULAR SHORTMP
27 TCF SHORTMP +1 # C(A) +0, DO REGULAR SHORTMP
28 TCF -2 # C(A) -, DO REGULAR SHORTMP
29 CS ZERO # C(A) -0, FORCE RESULT TO -0 AND RETURN.
30 TS MPAC
31 TS MPAC +1
32 TS MPAC +2
33 TC Q
34
35 FLASHON CAF BIT6 # TURN ON V/N FLASH
36 EXTEND # BIT 6 OF CHANNEL 11
37 WOR DSALMOUT
38 TC Q
39
40 FLASHOFF CS BIT6 # TURN OFF V/N FLASH
41 EXTEND
42 WAND DSALMOUT
43 TC Q
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```



```
1 # INTERNAL USE OF KEYBOARD AND DISPLAY PROGRAM.
2 #
3 #
4 # USER MUST SCHEDULE CALLS TO NVSUB SO THAT THERE IS NO CONFLICT OF USE OR
5 # CONFUSION TO OPERATOR. THE OLD GRABLOCK (INTERNAL/INTERNAL INTERLOCK)
6 # HAS BEEN REMOVED AND THE INTERNAL USER NO LONGER HAS THE PROTECTION THIS
7 # OFFERED.
8 #
9 # THERE ARE TWO WAYS A JOB CAN BE PUT TO SLEEP BY THE KEYBOARD + DISPLAY
10 # PROGRAM. 1) BY ENDIDLE
11 # 2) BY NVSUBUSY
12 # THE BASIC CONVENTION IS THAT ONLY ONE JOB WILL BE PERMITTED ASLEEP VIA
13 # THE KEYBOARD + DISPLAY PROGRAM AT A TIME. IF A JOB ATTEMPTS TO GO TO
14 # SLEEP BY MEANS OF (1) OR (2) AND THERE IS ALREADY A JOB ASLEEP THAT WAS
15 # PUT TO SLEEP BY (1) OR (2), THEN AN ABORT IS CAUSED.
16 #
17 # THE CALLING SEQUENCE FOR NVSUB IS
18 # CAF V/N
19 # L TC NVSUB
20 # L+1 RETURN HERE IF OPERATOR HAS INTERVENED
21 # L+2 RETURN HERE AFTER EXECUTION
22 #
23 # A ROUTINE CALLED NVSUBUSY IS PROVIDED (USE IS OPTIONAL) TO PUT
24 # YOUR JOB TO SLEEP UNTIL THE OPERATOR RELEASES THE KEYBOARD + DISPLAY
25 # SYSTEM. NVSUBUSY ALSO TURNS ON THE KEY RELEASE LIGHT.
26 # NVSUBUSY CANNOT BE CALLED FROM ERASABLE OR F/F MEMORY,
27 # SINCE JOBSLEEP AND JOBWAKE CAN HANDLE ONLY FIXED BANKS.
28 #
29 # THE CALLING SEQUENCE IS
30 # CAF WAKEFCADR
31 # TC NVSUBUSY
32 #
33 #
34 # .
35 #
36 # NVSUBUSY IS INTENDED FOR USE WHEN AN INTERNAL PROGRAM FINDS THE OPERATOR
37 # IS NOT USING THE KEYBOARD + DISPLAY PROGRAM (BY HIS OWN INITIATION). IT IS
38 # NOT INTENDED FOR USE WHEN ONE INTERNAL PROGRAM FINDS ANOTHER INTERNAL
39 # PROGRAM USING THE KEYBOARD + DISPLAY PROGRAM.
40 #
41 # NVSUBUSY ABORTS (WITH CODE 01206) IF A SECOND JOB ATTEMPTS TO GO TO
42 # SLEEP IN PINBALL. IN PARTICULAR, IF AN ATTEMPT IS MADE TO GO TO NVSUBUSY
43 # WHEN
44 # 1) DSPLIST NOT= +0. THIS IS THE CASE WHERE THE CAPACITY OF THE DSPLIST
45 # IS EXCEEDED.
46 # 2) CADRSTOR NOT= +0. THIS INDICATES THAT A JOB IS ALREADY USING
```

# ENDIDLE. (+-NZ INDICATE A JOB IS ALREADY ASLEEP DUE TO ENDIDLE.)

|          |      |          |                                           |
|----------|------|----------|-------------------------------------------|
| PRENVBSY | CS   | 2K+3     | # SPECIAL ENTRANCE FOR ROUTINES IN FIXED  |
|          | AD   | Q        | # BANKS ONLY DESIRING THE FCADR OF (LOC   |
|          | AD   | FBANK    | # FROM WHICH THE TC PRENVBSY WAS DONE) -2 |
| NVSUBUSY | TC   | POSTJUMP | # TO BE ENTERED.                          |
|          | CADR | NVSUBSY1 |                                           |
| 2K+3     | OCT  | 2003     |                                           |

# NVSUBSY1 MUST BE IN BANK 27 OR LOWER, SO IT WILL PUT CALLER TO SLEEP  
# WITH HIS PROPER SUPERBITS.

|          |        |            |                              |
|----------|--------|------------|------------------------------|
|          | SETLOC | ENDSPMM +1 |                              |
|          | COUNT* | \$\$/PIN   |                              |
| NVSUBSY1 | TS     | L          |                              |
|          | TC     | ISCADR+0   | # ABORT IF CADRSTOR NOT= +0. |
|          | TC     | ISLIST+0   | # ABORT IF DSPLIST NOT= +0.  |
|          | TC     | RELDSPON   |                              |
|          | CA     | L          |                              |
|          | TS     | DSPLIST    |                              |
| ENDNVBSY | TC     | JOBSLEEP   |                              |

# NVSBWAIT IS A SPECIAL ENTRANCE FOR ROUTINES IN FIXED BANKS ONLY. IF  
# SYSTEM IS NOT BUSY, IT EXECUTES V/N AND RETURNS TO L+1 (L= LOC FROM  
# WHICH THE TC NVSBWAIT WAS DONE). IF SYSTEM IS BUSY, IT PUTS CALLING JOB  
# TO SLEEP WITH L-1 GOING INTO LIST FOR EVENTUAL WAKING UP WHEN SYSTEM  
# IS NOT BUSY.

|          |        |             |                                            |
|----------|--------|-------------|--------------------------------------------|
|          | SETLOC | NVSUBUSY +3 |                                            |
|          | COUNT* | \$\$/PIN    |                                            |
| NVSBWAIT | LXCH   | 7           | # ZERO NVMONOPT OPTIONS                    |
|          | TS     | NVTEMP      |                                            |
|          | CAF    | BIT14       |                                            |
|          | MASK   | MONSAVE1    | # EXTERNAL MONITOR BIT                     |
|          | AD     | DSPLOCK     |                                            |
|          | CCS    | A           |                                            |
|          | TCF    | NVSBWT1     | # BUSY                                     |
|          | TCF    | NVSBKOM     | # FREE. NVSUB WILL SAVE L+1 FOR RETURN     |
|          |        |             | # AFTER EXECUTION.                         |
| NVSBWT1  | INCR   | Q           | # L+2. PRENVBSY WILL PUT L-1 INTO LIST AND |
|          | TCF    | PRENVBSY    | # GO TO SLEEP.                             |

# RELDSP IS USED BY VBPROC, VBTERM, VBRQEXEC, VBRQWAIT, VBRELDSP, EXTENDED  
# VERB DISPATCHER, VBRESEQ, RECALTST.  
# RELDSP1 IS USED BY MONITOR SET UP, VBRELDSP.

|        |     |        |                                        |
|--------|-----|--------|----------------------------------------|
| RELDSP | XCH | Q      | # SET DSPLOCK TO +0, TURN RELDSP LIGHT |
|        | TS  | RELRET | # OFF, SEARCH DSPLIST                  |
|        | CS  | BIT14  |                                        |

|    |               |               |                                            |    |
|----|---------------|---------------|--------------------------------------------|----|
| 1  |               |               |                                            | 1  |
| 2  | INHINT        |               |                                            | 2  |
| 3  | MASK MONSAVE1 |               |                                            | 3  |
| 4  |               | TS MONSAVE1   | # TURN OFF EXTERNAL MONITOR BIT            | 5  |
| 5  |               | CCS DSPLIST   |                                            | 6  |
| 6  |               | TC +2         |                                            | 7  |
| 7  |               | TC RELDSP2    | # LIST EMPTY                               | 9  |
| 8  |               | CAF ZERO      |                                            | 10 |
| 9  |               | XCH DSPLIST   |                                            | 11 |
| 10 |               | TC JOBWAKE    |                                            | 13 |
| 11 | RELDSP2       | RELINT        |                                            | 14 |
| 12 |               | CS BIT5       | # TURN OFF KEY RELEASE LIGHT               | 15 |
| 13 |               | EXTEND        | # (BIT 5 OF CHANNEL 11)                    | 17 |
| 14 |               | WAND DSALMOUT |                                            | 18 |
| 15 |               | CAF ZERO      |                                            | 19 |
| 16 |               | TS DSPLOCK    |                                            | 21 |
| 17 |               | TC RELRET     |                                            | 22 |
| 18 | RELDSP1       | XCH Q         | # SET DSPLOCK TO +0. NO DSPLIST SEARCH.    | 23 |
| 19 |               | TS RELRET     | # TURN KEY RLSE LIGHT OFF IF DSPLIST IS    | 25 |
| 20 |               |               | # EMPTY. LEAVE KEY RLSE LIGHT ALONE IF     | 26 |
| 21 |               |               | # DSPLIST IS NOT EMPTY.                    | 27 |
| 22 |               | CCS DSPLIST   |                                            | 29 |
| 23 |               | TC +2         | # + NOT EMPTY. LEAVE KEY RLSE LIGHT ALONE. | 30 |
| 24 |               | TC RELDSP2    | # +0 EMPTY. TURN OFF KEY RLSE LIGHT        | 31 |
| 25 |               | CAF ZERO      | # - NOT EMPTY. LEAVE KEY RLSE LIGHT ALONE  | 33 |
| 26 |               | TS DSPLOCK    |                                            | 34 |
| 27 |               | TC RELRET     |                                            | 35 |
| 28 |               |               |                                            | 36 |
| 29 | ENDPINBF      | EQUALS        |                                            | 37 |
| 30 |               |               |                                            | 38 |
| 31 |               |               |                                            | 39 |
| 32 |               |               |                                            | 40 |
| 33 |               |               |                                            | 41 |
| 34 |               |               |                                            | 42 |
| 35 |               |               |                                            | 43 |
| 36 |               |               |                                            | 44 |
| 37 |               |               |                                            | 45 |
| 38 |               |               |                                            | 46 |
| 39 |               |               |                                            | 47 |
| 40 |               |               |                                            | 48 |
| 41 |               |               |                                            | 49 |
| 42 |               |               |                                            | 50 |
| 43 |               |               |                                            | 51 |
| 44 |               |               |                                            | 52 |
| 45 |               |               |                                            | 53 |
| 46 |               |               |                                            | 54 |
| 47 |               |               |                                            | 55 |
| 48 |               |               |                                            | 56 |
| 49 |               |               |                                            | 57 |
| 50 |               |               |                                            | 58 |
| 51 |               |               |                                            | 59 |
| 52 |               |               |                                            | 60 |
| 53 |               |               |                                            | 61 |
| 54 |               |               |                                            | 62 |
| 55 |               |               |                                            | 63 |
| 56 |               |               |                                            | 64 |
| 57 |               |               |                                            | 65 |
| 58 |               |               |                                            | 66 |
| 59 |               |               |                                            | 67 |
| 60 |               |               |                                            | 68 |



# PINTEST IS NEEDED FOR AUTO CHECK OF PINBALL.

PINTEST           EQUALS   LST2FAN

# VBTSTLTS TURNS ON ALL DISPLAY PANEL LIGHTS. AFTER 5 SEC, IT TURNS  
# OFF THE CAUTION AND STATUS LIGHTS.

SETLOC ENDNVSB1 +1

VBTSTLTS      COUNT\*    \$\$/PIN  
              INHINT  
              CS        BIT1            # SET BIT 1 OF IMODES33 SO IMUMON WONT  
              MASK     IMODES33        # TURN OUT ANY LAMPS.  
              AD        BIT1  
              TS        IMODES33

CAF        TSTCON1            # TURN ON UPLINK ACTIVITY, TEMP, KEY RLSE,  
EXTEND                        # V/N FLASH, OPERATOR ERROR.

WOR        DSALMOUT  
CAF        TSTCON2            # TURN ON NO ATT, GIMBAL LOCK, TRACKER,  
TS        DSPTAB +11D        # PROG ALM.

CAF        BIT10            # TURN ON TEST ALARM OUTBIT

EXTEND

WOR        CHAN13

TSTLTS1      CAF        TEN  
              TS        ERCNT  
              CS        FULLDSP

INDEX      ERCNT

TS        DSPTAB

CCS        ERCNT

TC        TSTLTS1

CS        FULLDSP1

TS        DSPTAB +1            # TURN ON 3 PLUS SIGNS

TS        DSPTAB +4

TS        DSPTAB +6

CAF        ELEVEN

TS        NOUT

RELINT

CAF        SHOLTS

INHINT

TC        WAITLIST

EBANK=     DSPTAB

2CADR     TSTLTS2

TC        ENDOFJOB            # DSPLOCK IS LEFT BUSY (FROM KEYBOARD  
                              # ACTION) UNTIL TSTLTS3 TO INSURE THAT  
                              # LIGHTS TEST WILL BE SEEN.

FULLDSP      OCT        05675            # DISPLAY ALL 8'S  
FULLDSP1     OCT        07675            # DISPLAY ALL 8'S AND +  
TSTCON1      OCT        00175

# UPLINK ACTIVITY, TEMP. KEY RLSE,  
# V/N FLASH, OPERATOR ERROR.

|         |                          |                                             |                                                                                                                                           |
|---------|--------------------------|---------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| TSTCON2 | OCT                      | 40674                                       | # DSPTAB+11D BITS 3,4,5,6,8,9 LR LITES,<br># NO ATT, GIMBAL LOCK, TRACKER, PROG ALM.                                                      |
| TSTCON3 | OCT                      | 00115                                       | # CHAN 11 BITS 1, 3, 4, 7.<br># UPLINK ACITIVY, TEMP, OPERATOR ERROR.                                                                     |
| SHOLTS  | OCT                      | 764                                         | # 5 SEC                                                                                                                                   |
| TSTLTS2 | CAF<br>TC                | CHRPRI0<br>NOVAC                            | # CALLED BY WAITLIST                                                                                                                      |
|         | EBANK=<br>2CADR          | DSPTAB<br>TSTLTS3                           |                                                                                                                                           |
|         | TC                       | TASKOVER                                    |                                                                                                                                           |
| TSTLTS3 | CS                       | TSTCON3                                     | # CALLED BY EXECUTIVE                                                                                                                     |
|         | INHINT<br>EXTEND<br>WAND | DSALMOUT                                    | # TURN OFF UPLINK ACTIVITY, TEMP,<br># OPERATOR ERROR.                                                                                    |
|         | CS<br>EXTEND<br>WAND     | BIT10<br>CHAN13                             | # TURN OFF TEST ALARM OUTBIT                                                                                                              |
|         | CAF<br>EXTEND<br>RAND    | BIT4<br>CHAN12                              | # MAKE NO ATT FOLLOW BIT 4 OF CHANNEL 12<br># (NO ATT LIGHT ON IF IN COARSE ALIGN)                                                        |
|         | AD<br>TS<br>CS           | BIT15<br>DSPTAB +11D<br>13-11,1             | # TURN OFF AUTO, HOLD, FREE, SPARE,<br># GIMBAL LOCK, SPARE, TRACKER, PROG ALM<br># SET BITS TO INDICATE ALL LAMPS OUT. TEST              |
|         | MASK<br>AD<br>TS         | IMODES33<br>PRI016<br>IMODES33              | # LIGHTS COMPLETE.                                                                                                                        |
|         | CS<br>MASK               | OCT55000<br>IMODES30                        |                                                                                                                                           |
|         | AD<br>TS                 | PRI015<br>IMODES30                          | # 15000.                                                                                                                                  |
|         | CS<br>MASK<br>AD<br>TS   | RFAILS2<br>RADMODES<br>RCDUFBIT<br>RADMODES |                                                                                                                                           |
|         | RELINT                   |                                             |                                                                                                                                           |
|         | TC<br>CADR               | BANKCALL<br>DSPMM                           | # REDISPLAY C(MODREG)                                                                                                                     |
|         | TC<br>TC<br>TC<br>CADR   | KILMONON<br>FLASHOFF<br>POSTJUMP<br>TSTLTS4 | # TURN ON KILL MONITOR BIT.<br># TURN OFF V/N FLASH.<br># DOES RELDSP AND GOES TO PINBRNCH IF<br># ENDIDLE IS AWAITING OPERATOR RESPONSE. |

|          |        |       |                                  |
|----------|--------|-------|----------------------------------|
| 13-11,1  | OCT    | 16001 |                                  |
| RFAILS2  | OCT    | 330   | # RADAR CDU AND DATA FAIL FLAGS. |
| OCT55000 | OCT    | 55000 |                                  |
| ENDPINS2 | EQUALS |       |                                  |

```
1 # ERROR LIGHT RESET (RSET) TURNS OFF:
2 # UPLINK ACTIVITY, AUTO, HOLD, FREE, OPERATOR ERROR,
3 # PROG ALM, TRACKER FAIL.
4 # LEAVES GIMBAL LOCK AND NO ATT ALONE.
5 # IT ALSO ZEROS THE 'TEST ALARM' OUT BIT, WHICH TURNS OFF STBY, RESTART.
6 # IT ALSO SETS 'CAUTION RESET' TO 1.
7 # IT ALSO FORCES BIT 12 OF ALL DSPTAB ENTRIES TO 1.
```

|         |        |             |                                           |
|---------|--------|-------------|-------------------------------------------|
|         | SETLOC | DOPROC +2   |                                           |
|         | COUNT* | \$\$/PIN    |                                           |
| ERROR   | XCH    | 21/22REG    | # RESTORE ORIGINAL C(DSPLOCK). THUS ERROR |
|         | TS     | DSPLOCK     | # LIGHT RESET LEAVES DSPLOCK UNCHANGED.   |
|         | INHINT |             |                                           |
|         | CAF    | BIT10       | # TURN ON 'CAUTION RESET' OUTBIT          |
|         | EXTEND |             |                                           |
|         | WOR    | DSALMOUT    | # BIT10 CHAN 11                           |
|         | CAF    | GL+NOATT    | # LEAVE GIMBAL LOCK AND NO ATT INTACT,    |
|         | MASK   | DSPTAB +11D | # TURNING OFF AUTO, HOLD, FREE,           |
|         | AD     | BIT15       | # PROG ALARM, AND TRACKER.                |
|         | TS     | DSPTAB +11D |                                           |
|         | CS     | PRI016      | # RESET FAIL BITS WHICH GENERATE PROG     |
|         | MASK   | IMODES33    | # ALARM SO THAT IF THE FAILURE STILL      |
|         | AD     | PRI016      | # EXISTS, THE ALARM WILL COME BACK.       |
|         | TS     | IMODES33    |                                           |
|         | CS     | BIT10       |                                           |
|         | MASK   | IMODES30    |                                           |
|         | AD     | BIT10       |                                           |
|         | TS     | IMODES30    |                                           |
|         | CS     | RFAILS      |                                           |
|         | MASK   | RADMODES    |                                           |
|         | AD     | RCDUFBIT    |                                           |
|         | TS     | RADMODES    |                                           |
|         | CS     | BIT10       | # TURN OFF 'TEST ALARM' OUTBIT.           |
|         | EXTEND |             |                                           |
|         | WAND   | CHAN13      |                                           |
|         | CS     | ERCON       | # TURN OFF UPLINK ACTIVITY,               |
|         | EXTEND |             | # OPERATOR ERROR.                         |
|         | WAND   | DSALMOUT    |                                           |
| TSTAB   | CAF    | BINCON      | # (DEC 10)                                |
|         | TS     | ERCNT       | # ERCNT = COUNT                           |
|         | INHINT |             |                                           |
|         | INDEX  | ERCNT       |                                           |
|         | CCS    | DSPTAB      |                                           |
|         | AD     | ONE         |                                           |
|         | TC     | ERPLUS      |                                           |
|         | AD     | ONE         |                                           |
| ERMINUS | CS     | A           |                                           |
|         | MASK   | NOTBIT12    |                                           |





|    |          |        |            |                                         |    |
|----|----------|--------|------------|-----------------------------------------|----|
| 1  |          |        |            |                                         | 1  |
| 2  |          | TC     | ERCOM      |                                         | 2  |
| 3  | ERPLUS   | CS     | A          |                                         | 3  |
| 4  |          | MASK   | NOTBIT12   |                                         | 4  |
| 5  |          | CS     | A          | # MIGHT WANT TO RESET CLPASS, DECBRNCH, | 5  |
| 6  | ERCOM    | INDEX  | ERCNT      | # ETC.                                  | 6  |
| 7  |          | TS     | DSPTAB     |                                         | 7  |
| 8  |          | RELINT |            |                                         | 8  |
| 9  |          | CCS    | ERCNT      |                                         | 9  |
| 10 |          | TC     | TSTAB +1   |                                         | 10 |
| 11 |          | CAF    | ZERO       |                                         | 11 |
| 12 |          | TS     | FAILREG    |                                         | 12 |
| 13 |          | TS     | FAILREG +1 |                                         | 13 |
| 14 |          | TS     | FAILREG +2 |                                         | 14 |
| 15 |          | TS     | SFAIL      |                                         | 15 |
| 16 |          | TC     | ENDOFJOB   |                                         | 16 |
| 17 |          |        |            |                                         | 17 |
| 18 | ERCON    | OCT    | 104        | # CHAN 11 BITS 3,7.                     | 18 |
| 19 |          |        |            | # UPLINK ACTIVITY, AND OPERATOR ERROR.  | 19 |
| 20 | RFAILS   | OCT    | 330        | # RADAR CDU AND DATA FAIL FLAGS.        | 20 |
| 21 | GL+NOATT | OCT    | 00050      | # NO ATT AND GIMBAL LOCK LAMPS          | 21 |
| 22 | NOTBIT12 | OCT    | 73777      |                                         | 22 |
| 23 |          |        |            |                                         | 23 |
| 24 | ENDPINS1 | EQUALS |            |                                         | 24 |
| 25 |          |        |            |                                         | 25 |
| 26 |          | SBANK= | LOWSUPER   |                                         | 26 |
| 27 |          |        |            |                                         | 27 |
| 28 |          |        |            |                                         | 28 |
| 29 |          |        |            |                                         | 29 |
| 30 |          |        |            |                                         | 30 |
| 31 |          |        |            |                                         | 31 |
| 32 |          |        |            |                                         | 32 |
| 33 |          |        |            |                                         | 33 |
| 34 |          |        |            |                                         | 34 |
| 35 |          |        |            |                                         | 35 |
| 36 |          |        |            |                                         | 36 |
| 37 |          |        |            |                                         | 37 |
| 38 |          |        |            |                                         | 38 |
| 39 |          |        |            |                                         | 39 |
| 40 |          |        |            |                                         | 40 |
| 41 |          |        |            |                                         | 41 |
| 42 |          |        |            |                                         | 42 |
| 43 |          |        |            |                                         | 43 |
| 44 |          |        |            |                                         | 44 |
| 45 |          |        |            |                                         | 45 |
| 46 |          |        |            |                                         | 46 |
| 47 |          |        |            |                                         | 47 |
| 48 |          |        |            |                                         | 48 |
| 49 |          |        |            |                                         | 49 |
| 50 |          |        |            |                                         | 50 |
| 51 |          |        |            |                                         | 51 |
| 52 |          |        |            |                                         | 52 |
| 53 |          |        |            |                                         | 53 |
| 54 |          |        |            |                                         | 54 |
| 55 |          |        |            |                                         | 55 |
| 56 |          |        |            |                                         | 56 |
| 57 |          |        |            |                                         | 57 |
| 58 |          |        |            |                                         | 58 |
| 59 |          |        |            |                                         | 59 |
| 60 |          |        |            |                                         | 60 |

```
1 # MOD NO: 0 DATE: 1 MAY 1968
2 # MOD BY: DIGITAL DEVEL GROUP LOG SECTION R60,R62
3
4 #
5 # FUNCTIONAL DESCRIPTION:
6 #
7 # CALLED AS A GENERAL SUBROUTINE TO MANEUVER THE LM TO A SPECIFIED
8 # ATTITUDE.
9 #
10 # 1. IF THE 3-AXIS FLAG IS NOT SET THE FINAL CDU ANGLES ARE
11 # CALCULATED (VECPPOINT).
12 #
13 # 2. THE FDAI BALL ANGLES (NOUN 18) ARE CALCULATED (BALLANGS).
14 #
15 # 3. REQUEST FLASHING DISPLAY V50 N18 PLEASE PERFORM AUTO MANEUVER.
16 #
17 # 4. IF PRIORITY DISPLAY FLAG IS SET DO A PHASECHANGE. THEN AWAIT
18 # ASTRONAUT RESPONSE.
19 #
20 # 5. DISPLAY RESPONSE RETURNS:
21 #
22 # A. ENTER - RESET 3-AXIS FLAG AND RETURN TO CLIENT.
23 #
24 # B. TERMINATE - IF IN P00 GO TO STEP 5A. OTHERWISE CHECK IF R61 IS
25 # THE CALLING PROGRAM. IF IN R61 AN EXIT IS MADE TO GOTOV56. IF
26 # NOT IN R61 AN EXIT IS DONE VIA GOTOPOOH.
27 #
28 # C. PROCEED - CONTINUE WITH PROGRAM AT STEP 6.
29 #
30 # 6. IF THE 3-AXISFLAG IS NOT SET, THE FINAL CDU ANGLES ARE CALCULATED
31 # (VECPPOINT).
32 #
33 # 7. THE FDAI BALL ANGLES (NOUN 18) ARE CALCULATED (BALLANGS).
34 #
35 # 8. IF THE G+N SWITCH IS NOT SET GO BACK TO STEP 3.
36 #
37 # 9. IF THE AUTO SWITCH IS NOT SET GO BACK TO STEP 3.
38 #
39 # 10. NONFLASHING DISPLAY V06N18 (FDAI ANGLES).
40 #
41 # 11. DO A PHASECHANGE.
42 #
43 # 12. DO A MANEUVER CALCULATION AND ICDU DRIVE ROUTINE TO ACHIEVE FINAL
44 #
45 # GIMBAL ANGLES (GOMANUR).
46 #
47 # 13. AT END OF MANEUVER GO TO STEP 3.
48 #
49 # IF SATISFACTORY MANEUVER STEP 5A EXITS R60.
50 # FOR FURTHER ADJUSTMENT OF THE VEHICLE ATTITUDE ABOUT THE
51 # DESIRED VECTOR, THE ROUTINE MAY BE PERFORMED AGAIN STARTING AT
```

# STEP 5C.

# CALLING SEQUENCE: TC BANKCALL  
# CADR R60LEM# ERASABLE INITIALIZATION REQUIRED : SCAXIS, POINTVSM (FOR VECPOINT)  
# 3AXISFLG.# SUBROUTINES CALLED: VECPOINT, BALLANGS, GOPERF2R, LINUS, GODSPER,  
# GOMANUR, DOWNFLAG, PHASCHNG, UPFLAG# NORMAL EXIT MODES: CAE TEMPR60 (CALLERS RETURN ADDRESS)  
# TC BANKJUMP

# ALARMS: NONE

# OUTPUT: NONE

# DEBRIS: CPHI, CTHETA, CPSI, 3AXISFLG, TBASE2

BANK 34  
SETLOC MANUVER  
BANK

EBANK= TEMPR60

R60LEM COUNT\* \$\$/R06  
TC MAKECADR  
TS TEMPR60REDOMANN CAF 3AXISBIT  
MASK FLAGWRD5 # IS 3-AXIS FLAG SETCCS A  
TCF TOBALL # YES  
TC INTPRETCALL  
STORE VECPOINT # TO COMPUTE FINAL ANGLES  
CPHI # STORE FINAL ANGLES - CPHI,CTHETA,CPSI  
EXIT

TOBALL TC BANKCALL

TOBALLA CADR BALLANGS # TO CONVERT ANGLES TO FDAI  
CAF V06N18TC BANKCALL  
CADR GOPERF2R # DISPLAY PLEASE PERFORM AUTO MANEUVER  
TC R61TEST  
TC REDOMANC # PROCEED  
TC ENDMANU1 # ENTER I.E. FINISHED WITH R60

|          |       |           |                                         |
|----------|-------|-----------|-----------------------------------------|
|          | TC    | CHKLINUS  | # TO CHECK FOR PRIORITY DISPLAYS        |
|          | TC    | ENDOFJOB  |                                         |
| REDOMANC | CAF   | 3AXISBIT  |                                         |
|          | MASK  | FLAGWRD5  | # IS 3-AXIS FLAG SET                    |
|          | CCS   | A         |                                         |
|          | TCF   | TOBALLC   | # YES                                   |
|          | TC    | INTPRET   |                                         |
|          | CALL  |           |                                         |
|          |       | VECPPOINT | # TO COMPUTE FINAL ANGLES               |
|          | STORE | CPHI      | # STORE ANGLES                          |
|          | EXIT  |           |                                         |
| TOBALLC  | TC    | BANKCALL  |                                         |
|          | CADR  | BALLANGS  | # TO CONVERT ANGLES TO FDAI             |
|          | TC    | G+N,AUTO  | # CHECK AUTO MODE                       |
|          | CCS   | A         |                                         |
|          | TCF   | TOBALLA   | # NOT AUTO, GO REREQUEST AUTO MANEUVER. |
| AUTOMANV | CAF   | V06N18    | # STATIC DISPLAY DURING AUTO MANEUVER   |
|          | TC    | BANKCALL  |                                         |
|          | CADR  | GODSPR    |                                         |
|          | TC    | CHKLINUS  | # TO CHECK FOR PRIORITY DISPLAYS        |
| STARTMNV | TC    | BANKCALL  | # PERFORM MANEUVER VIA KALCMANU         |
|          | CADR  | GOMANUR   |                                         |
| ENDMANUV | TCF   | TOBALLA   | # FINISHED MANEUVER.                    |
| ENDMANU1 | TC    | DOWNFLAG  | # RESET 3-AXIS FLAG                     |
|          | ADRES | 3AXISFLG  |                                         |
|          | CAE   | TEMPR60   |                                         |
|          | TC    | BANKJUMP  |                                         |
| CHKLINUS | CS    | FLAGWRD4  |                                         |
|          | MASK  | PDSPFBIT  | # IS PRIORITY DISPLAY FLAG SET?         |
|          | CCS   | A         |                                         |
|          | TC    | Q         | # NO - EXIT                             |
|          | CA    | Q         |                                         |
|          | TS    | MPAC +2   | # SAVE RETURN                           |
|          | CS    | THREE     | # OBTAIN LOCATION FOR RESTART           |
|          | AD    | BUF2      | # HOLDS Q OF LAST DISPLAY               |
|          | TS    | TBASE2    |                                         |
|          | TC    | PHASCHNG  |                                         |
|          | OCT   | 00132     |                                         |
|          | CAF   | BIT7      |                                         |
|          | TC    | LINUS     | # GO SET BITS FOR PRIORITY DISPLAY      |
|          | TC    | MPAC +2   |                                         |

|                                                           |                      |                           |                                           |
|-----------------------------------------------------------|----------------------|---------------------------|-------------------------------------------|
| RELINUS                                                   | CAF<br>TC            | PRI026<br>PRIOCHNG        | # RESTORE ORIGINAL PRIORITY               |
|                                                           | CAF<br>MASK          | TRACKBIT<br>FLAGWRD1      | # DON'T CONTINUE R60 UNLESS TRACKFLAG ON. |
|                                                           | CCS<br>TCF           | A<br>RER60                |                                           |
|                                                           | CAF<br>MASK<br>CCS   | RNDVZBIT<br>FLAGWRD0<br>A | # IS IT P20?                              |
|                                                           | TC<br>TC<br>OCT      | +4<br>PHASCHNG<br>40112   | # YES<br># NO, MUST BE P25, SET 2.11 SPOT |
|                                                           | TC                   | ENDOFJOB                  |                                           |
|                                                           | TC<br>OCT            | PHASCHNG<br>40072         | # SET 2.7 SPOT FOR P20                    |
|                                                           | TC                   | ENDOFJOB                  |                                           |
| RER60                                                     | TC<br>ADRES          | UPFLAG<br>PDSPFLAG        | # SET PRIO DISPLAY FLAG AFTER RESTART     |
|                                                           | TC                   | TBASE2                    |                                           |
| R61TEST                                                   | CA<br>EXTEND         | MODREG                    | # IF WE ARE IN P00 IT MUST BE V49 OR V89  |
|                                                           | BZF                  | ENDMANU1                  | # THUS WE GO TO ENDEXT VIA USER           |
|                                                           | CA<br>MASK<br>EXTEND | FLAGWRD4<br>PDSPFBIT      | # ARE WE IN R61 (P20 OR P25)              |
|                                                           | BZF<br>TC            | GOTOP00H<br>GOTOV56       | # NO<br># YES                             |
| BIT14+7                                                   | OCT                  | 20100                     |                                           |
| OCT203                                                    | OCT                  | 203                       |                                           |
| V06N18                                                    | VN                   | 0618                      |                                           |
| # SUBROUTINE TO CHECK FOR G+N CONTROL. AUTO STABILIZATION |                      |                           |                                           |
| #                                                         |                      |                           |                                           |
| # RETURNS WITH C(A) = + IF NOT SET FOR G+N, AUTO          |                      |                           |                                           |
| # RETURNS WITH C(A) = +0 IF SWITCHES ARE SET              |                      |                           |                                           |
| G+N,AUTO                                                  | EXTEND               |                           |                                           |
|                                                           | READ<br>MASK<br>CCS  | CHAN30<br>BIT10<br>A      |                                           |
|                                                           | TC                   | Q                         | # NOT IN G+N C(A) = +                     |



|    |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  |    |  |  |   |
|----|---------------------------------|------------------------------|----------------------|-----------------------|--|--|--|--|--|--|--|--|--|--|--|--|----|--|--|---|
| 1  | ISITAUTO                        | EXTEND<br>READ<br>MASK<br>TC | CHAN31<br>BIT14<br>Q | # CHECK FOR AUTO MODE |  |  |  |  |  |  |  |  |  |  |  |  |    |  |  | 1 |
| 2  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  |    |  |  | 2 |
| 3  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  |    |  |  | 3 |
| 4  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 4  |  |  |   |
| 5  | # (+) = NOT IN AUTO, (+0) = AOK |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 5  |  |  |   |
| 6  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 6  |  |  |   |
| 7  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 7  |  |  |   |
| 8  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 8  |  |  |   |
| 9  |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 9  |  |  |   |
| 10 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 10 |  |  |   |
| 11 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 11 |  |  |   |
| 12 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 12 |  |  |   |
| 13 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 13 |  |  |   |
| 14 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 14 |  |  |   |
| 15 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 15 |  |  |   |
| 16 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 16 |  |  |   |
| 17 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 17 |  |  |   |
| 18 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 18 |  |  |   |
| 19 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 19 |  |  |   |
| 20 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 20 |  |  |   |
| 21 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 21 |  |  |   |
| 22 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 22 |  |  |   |
| 23 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 23 |  |  |   |
| 24 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 24 |  |  |   |
| 25 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 25 |  |  |   |
| 26 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 26 |  |  |   |
| 27 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 27 |  |  |   |
| 28 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 28 |  |  |   |
| 29 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 29 |  |  |   |
| 30 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 30 |  |  |   |
| 31 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 31 |  |  |   |
| 32 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 32 |  |  |   |
| 33 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 33 |  |  |   |
| 34 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 34 |  |  |   |
| 35 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 35 |  |  |   |
| 36 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 36 |  |  |   |
| 37 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 37 |  |  |   |
| 38 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 38 |  |  |   |
| 39 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 39 |  |  |   |
| 40 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 40 |  |  |   |
| 41 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 41 |  |  |   |
| 42 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 42 |  |  |   |
| 43 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 43 |  |  |   |
| 44 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 44 |  |  |   |
| 45 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 45 |  |  |   |
| 46 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 46 |  |  |   |
| 47 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 47 |  |  |   |
| 48 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 48 |  |  |   |
| 49 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 49 |  |  |   |
| 50 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 50 |  |  |   |
| 51 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 51 |  |  |   |
| 52 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 52 |  |  |   |
| 53 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 53 |  |  |   |
| 54 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 54 |  |  |   |
| 55 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 55 |  |  |   |
| 56 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 56 |  |  |   |
| 57 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 57 |  |  |   |
| 58 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 58 |  |  |   |
| 59 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 59 |  |  |   |
| 60 |                                 |                              |                      |                       |  |  |  |  |  |  |  |  |  |  |  |  | 60 |  |  |   |

```
PROGRAM DESCRIPTION BALLANGS
MOD NO. LOG SECTION R60,R62

#
WRITTEN BY RAMA M.AIYAWAR
FUNCTIONAL DESCRIPTION
#
COMPUTES LM FDAI BALL DISPLAY ANGLES
CALLING SEQUENCE
#
TC BALLANGS
NORMAL EXIT MODE
#
TC BALLEXIT # (SAVED Q)
#
ALARM OR EXIT MODE NIL
SUBROUTINES CALLED
CD*TR*G
ARCTAN
#
INPUT
#
CPHI,CTHETA,CPSI ARE THE ANGLES CORRESPONDING TO AOG,AIG,AMG. THEY ARE
SP,2S COMPLIMENT SCALED TO HALF REVOLUTION.
OUTPUT
#
FDAIX,FDAIY,FDAIZ ARE THE REQUIRED BALL ANGLES SCALED TO HALF REVOLUTION
SP,2S COMPLIMENT.
THESE ANGLES WILL BE DISPLAYED AS DEGREES AND HUNDREDTHS. IN THE ORDER ROLL, PITCH, YAW, USING NOUNS 18 & 19.
#
ERASABLE INITIALIZATION REQUIRED
#
CPHI,CTHETA,CPSI EACH A SP REGISTER
DEBRIS
#
A,L,Q,MPAC,SINCDU,COSCDU,PUSHLIS,BALLEXIT
#
#
NOMENCLATURE: CPHI, CTHETA, & CPSI REPRESENT THE OUTER, INNER, & MIDDLE GIMBAL ANGLES, RESPECTIVELY; OR
EQUIVALENTLY, CDUX, CDUY, & CDUZ.
#
NOTE: ARCTAN CHECKS FOR OVERFLOW AND SHOULD BE ABLE TO HANDLE ANY SINGULARITIES.

 SETLOC BAWLANGS
 BANK

BALLANGS COUNT* $$/BALL
 TC MAKECADR
 TS BALLEXIT
 CA CPHI
```

|    |         |        |            |                                            |
|----|---------|--------|------------|--------------------------------------------|
| 1  |         |        |            |                                            |
| 2  |         | TS     | CDUSPOT +4 |                                            |
| 3  |         | CA     | CTHETA     |                                            |
| 4  |         | TS     | CDUSPOT    |                                            |
| 5  |         | CA     | CPSI       |                                            |
| 6  |         | TS     | CDUSPOT +2 |                                            |
| 7  |         |        |            |                                            |
| 8  |         | TC     | INTPRET    |                                            |
| 9  |         | SETPD  | CALL       |                                            |
| 10 |         |        | OD         |                                            |
| 11 |         |        | CD*TR*G    |                                            |
| 12 |         |        |            |                                            |
| 13 |         | DLOAD  | DMP        |                                            |
| 14 |         |        | SINCDUX    | # SIN (OGA)                                |
| 15 |         |        | COSCDUZ    | # COS (MGA)                                |
| 16 |         |        |            |                                            |
| 17 |         | SL1    | DCOMP      | # SCALE                                    |
| 18 |         | ARCSIN | PDDL       | # YAW = ARCSIN(-SXCZ) INTO 0 PD            |
| 19 |         |        | SINCDUZ    |                                            |
| 20 |         | STODL  | SINTH      | # (SINTH = 18D IN PD)                      |
| 21 |         |        | COSCDUZ    |                                            |
| 22 |         | DMP    | SL1        | # RESCALE                                  |
| 23 |         |        | COSCDUX    |                                            |
| 24 |         | STCALL | COSTH      | # (COSTH= 16D IN PD)                       |
| 25 |         |        | ARCTAN     |                                            |
| 26 |         | PDDL   | DMP        | # ROLL = ARCTAN(SZ/CZCX) INTO 2 PD         |
| 27 |         |        | SINCDUZ    |                                            |
| 28 |         |        | SINCDUX    |                                            |
| 29 |         | SL2    | PUSH       | # SXSZ INTO 4 PD                           |
| 30 |         | DMP    | PDDL       | # SXSZCY INTO 4 PD                         |
| 31 |         |        | COSCDUY    |                                            |
| 32 |         | DMP    | PDDL       | # SXSZSY INTO 6 PD                         |
| 33 |         |        | SINCDUY    |                                            |
| 34 |         |        | COSCDUX    |                                            |
| 35 |         | DMP    | SL1        | # CXCY                                     |
| 36 |         |        | COSCDUY    |                                            |
| 37 |         | DSU    | STADR      | # PULL UP FROM 6 PD                        |
| 38 |         | STODL  | COSTH      | # COSTH = CXCY - SXSZSY                    |
| 39 |         |        | SINCDUY    |                                            |
| 40 |         | DMP    | SL1        |                                            |
| 41 |         |        | COSCDUX    | # CXSY                                     |
| 42 |         | DAD    | STADR      | # PULL UP FROM 4 PD                        |
| 43 |         | STCALL | SINTH      | # SINTH = CXSY + SXSZCY                    |
| 44 |         |        | ARCTAN     | # RETURNS WITH D(MPAC) = PITCH             |
| 45 |         | PDDL   | VDEF       | # PITCH INTO 2 PD, ROLL INTO MPAC FROM 2PD |
| 46 |         | RTB    |            | # VDEF MAKES V(MPAC) = ROLL, PITCH, YAW    |
| 47 |         |        | V1STO2S    |                                            |
| 48 |         | STORE  | FDAIX      | # MODE IS TP                               |
| 49 |         | EXIT   |            |                                            |
| 50 |         |        |            |                                            |
| 51 | ENDBALL | CA     | BALLEXIT   |                                            |
| 52 |         |        |            |                                            |
| 53 |         |        |            |                                            |
| 54 |         |        |            |                                            |
| 55 |         |        |            |                                            |
| 56 |         |        |            |                                            |
| 57 |         |        |            |                                            |
| 58 |         |        |            |                                            |
| 59 |         |        |            |                                            |
| 60 |         |        |            |                                            |



TC                      BANKJUMP

## # PROGRAM DESCRIPTION - VECPOINT

#

#

# THIS INTERPRETIVE SUBROUTINE MAY BE USED TO POINT A SPACECRAFT AXIS IN A DESIRED DIRECTION. THE AXIS

# TO BE POINTED MUST APPEAR AS A HALF UNIT DOUBLE PRECISION VECTOR IN SUCCESSIVE LOCATIONS OF ERASABLE MEMORY

# BEGINNING WITH THE LOCATION CALLED SCAXIS. THE COMPONENTS OF THIS VECTOR ARE GIVEN IN SPACECRAFT COORDINATES.

# THE DIRECTION IN WHICH THIS AXIS IS TO BE POINTED MUST APPEAR AS A HALF UNIT DOUBLE PRECISION VECTOR IN

# SUCCESSIVE LOCATIONS OF ERASABLE MEMORY BEGINNING WITH THE ADDRESS CALLED POINTVSM. THE COMPONENTS OF THIS

# VECTOR ARE GIVEN IN STABLE MEMBER COORDINATES. WITH THIS INFORMATION VECPOINT COMPUTES A SET OF THREE GIMBAL

# ANGLES (2S COMPLEMENT) CORRESPONDING TO THE CROSS-PRODUCT ROTATION BETWEEN SCAXIS AND POINTVSM AND STORES THEM

# IN T(MPAC) BEFORE RETURNING TO THE CALLER.

# THIS ROTATION, HOWEVER, MAY BRING THE S/C INTO GIMBAL LOCK. WHEN POINTING A VECTOR IN THE Y-Z PLANE,

# THE TRANSPONDER AXIS, OR THE AOT FOR THE LEM, THE PROGRAM WILL CORRECT THIS PROBLEM BY ROTATING THE CROSS-

# PRODUCT ATTITUDE ABOUT POINTVSM BY A FIXED AMOUNT SUFFICIENT TO ROTATE THE DESIRED S/C ATTITUDE OUT OF GIMBAL

# LOCK. IF THE AXIS TO BE POINTED IS MORE THAN 40.6 DEGREES BUT LESS THAN 60.5 DEG FROM THE +X (OR-X) AXIS,

# THE ADDITIONAL ROTATION TO AVOID GIMBAL LOCK IS 35 DEGREES. IF THE AXIS IS MORE THAN 60.5 DEGREES FROM +X (OR -X)

# THE ADDITIONAL ROTATION IS 35 DEGREES. THE GIMBAL ANGLES CORRESPONDING TO THIS ATTITUDE ARE THEN COMPUTED AND

# STORED AS 2S COMPLIMENT ANGLES IN T(MPAC) BEFORE RETURNING TO THE CALLER.

# WHEN POINTING THE X-AXIS, OR THE THRUST VECTOR, OR ANY VECTOR WITHIN 40.6 DEG OF THE X-AXIS, VECPOINT

# CANNOT CORRECT FOR A CROSS-PRODUCT ROTATION INTO GIMBAL LOCK. IN THIS CASE A PLATFORM REALIGNMENT WOULD BE

# REQUIRED TO POINT THE VECTOR IN THE DESIRED DIRECTION. AT PRESENT NO INDICATION IS GIVEN FOR THIS SITUATION

# EXCEPT THAT THE FINAL MIDDLE GIMBAL ANGLE IN MPAC +2 IS GREATER THAN 59 DEGREES.

#

# CALLING SEQUENCE -

# 1) LOAD SCAXIS, POINTVSM

# 2) CALL

#

VECPOINT

#

#

RETURNS WITH

#

# 1) DESIRED OUTER GIMBAL ANGLE IN MPAC

# 2) DESIRED INNER GIMBAL ANGLE IN MPAC +1

# 3) DESIRED MIDDLE GIMBAL ANGLE IN MPAC +2

#

#

ERASABLES USED -

#

# 1) SCAXIS 6

# 2) POINTVSM 6

# 3) MIS 18

# 4) DEL 18

# 5) COF 6

# 6) VECQTEMP 1

# 7) ALL OF VAC AREA 43

#

# TOTAL 99

#

SETLOC VECPT

BANK

|    |          |        |            |                                          |
|----|----------|--------|------------|------------------------------------------|
| 1  |          | COUNT* | \$\$/VECPT |                                          |
| 2  |          |        |            |                                          |
| 3  |          |        |            |                                          |
| 4  |          | EBANK= | BCDU       |                                          |
| 5  |          |        |            |                                          |
| 6  | VECPNT1  | STQ    | BOV        | # THIS ENTRY USES DESIRED CDUS           |
| 7  |          |        | VECQTEMP   | # NOT PRESENT-ENTER WITH CDUD'S IN MPAC  |
| 8  |          |        | VECPNT2    |                                          |
| 9  | VECPNT2  | AXC,2  | GOTO       |                                          |
| 10 |          |        | MIS        |                                          |
| 11 |          |        | STORANG    |                                          |
| 12 | VECPOINT | STQ    | BOV        | # SAVE RETURN ADDRESS                    |
| 13 |          |        | VECQTEMP   |                                          |
| 14 |          |        | VECLEAR    | # AND CLEAR OVFIN                        |
| 15 | VECLEAR  | AXC,2  | RTB        |                                          |
| 16 |          |        | MIS        | # READ THE PRESENT CDU ANGLES AND        |
| 17 |          |        | READCDUK   | # STORE THEM IN PD25, 26, 27             |
| 18 | STORANG  | STCALL | 25D        |                                          |
| 19 |          |        | CDUTODCM   | # S/C AXES TO STABLE MEMBER AXES (MIS)   |
| 20 |          | VLOAD  | VXM        |                                          |
| 21 |          |        | POINTVSM   | # RESOLVE THE POINTING DIRECTION VF INTO |
| 22 |          |        | MIS        | # INITIAL S/C AXES ( VF = POINTVSM)      |
| 23 |          | UNIT   |            |                                          |
| 24 |          | STORE  | 28D        |                                          |
| 25 |          |        |            | # PD 28 29 30 31 32 33                   |
| 26 |          | VXV    | UNIT       | # TAKE THE CROSS PRODUCT VF X VI         |
| 27 |          |        | SCAXIS     | # WHERE VI = SCAXIS                      |
| 28 |          | BOV    | VCOMP      |                                          |
| 29 |          |        | PICKAXIS   |                                          |
| 30 |          | STODL  | COF        | # CHECK MAGNITUDE                        |
| 31 |          |        | 36D        | # OF CROSS PRODUCT                       |
| 32 |          | DSU    | BMN        | # VECTOR, IF LESS                        |
| 33 |          |        | DPB-14     | # THAN B-14 ASSUME                       |
| 34 |          |        | PICKAXIS   | # UNIT OPERATION                         |
| 35 |          | VLOAD  | DOT        | #                                        |
| 36 |          |        | SCAXIS     | INVALID.                                 |
| 37 |          |        | 28D        |                                          |
| 38 |          | SL1    | ARCCOS     |                                          |
| 39 | COMPMATX | CALL   |            | # NOW COMPUTE THE TRANSFORMATION FROM    |
| 40 |          |        | DELCOMP    | # FINAL S/C AXES TO INITIAL S/C AXES MFI |
| 41 |          | AXC,1  | AXC,2      |                                          |
| 42 |          |        | MIS        | # COMPUTE THE TRANSFORMATION FROM FINAL  |
| 43 |          |        | KEL        | # S/C AXES TO STABLE MEMBER AXES         |
| 44 |          | CALL   |            | # MFS = MIS MFI                          |
| 45 |          |        | MXM3       | # (IN PD LIST)                           |
| 46 |          |        |            |                                          |
| 47 |          | DLOAD  | ABS        |                                          |
| 48 |          |        | 6          | # MFS6 = SIN(CPSI) \$2                   |
| 49 |          | DSU    | BMN        |                                          |
| 50 |          |        | SINGIMLC   | # = SIN(59 DEGS) \$2                     |
| 51 |          |        | FINDGIMB   | # /CPSI/ LESS THAN 59 DEGS               |
| 52 |          |        |            |                                          |
| 53 |          |        |            |                                          |
| 54 |          |        |            |                                          |
| 55 |          |        |            |                                          |
| 56 |          |        |            |                                          |
| 57 |          |        |            |                                          |
| 58 |          |        |            |                                          |
| 59 |          |        |            |                                          |
| 60 |          |        |            |                                          |

|    |                                            |        |          |
|----|--------------------------------------------|--------|----------|
| 1  | # I.E. DESIRED ATTITUDE NOT IN GIMBAL LOCK |        |          |
| 2  |                                            |        |          |
| 3  |                                            |        |          |
| 4  |                                            | DLOAD  | ABS      |
| 5  |                                            |        | SCAXIS   |
| 6  |                                            | DSU    | BPL      |
| 7  |                                            |        | SINVEC1  |
| 8  |                                            |        | FINDGIMB |
| 9  |                                            | VLOAD  |          |
| 10 |                                            | STADR  |          |
| 11 |                                            | STOVL  | MIS +12D |
| 12 |                                            | STADR  |          |
| 13 |                                            | STOVL  | MIS +6   |
| 14 |                                            | STADR  |          |
| 15 |                                            | STOVL  | MIS      |
| 16 |                                            |        | MIS +6   |
| 17 |                                            | BPL    | VCOMP    |
| 18 |                                            |        | IGSAMEX  |
| 19 |                                            |        |          |
| 20 | IGSAMEX                                    | VXV    | BMN      |
| 21 |                                            |        | SCAXIS   |
| 22 |                                            |        | U=SCAXIS |
| 23 |                                            |        |          |
| 24 |                                            |        |          |
| 25 |                                            |        |          |
| 26 |                                            | VLOAD  | VCOMP    |
| 27 |                                            |        | SCAXIS   |
| 28 |                                            | STCALL | COF      |
| 29 |                                            |        | CHEKAXIS |
| 30 | U=SCAXIS                                   | VLOAD  |          |
| 31 |                                            |        | SCAXIS   |
| 32 |                                            | STORE  | COF      |
| 33 | CHEKAXIS                                   | DLOAD  | ABS      |
| 34 |                                            |        | SCAXIS   |
| 35 |                                            | DSU    | BPL      |
| 36 |                                            |        | SINVEC2  |
| 37 |                                            |        | PICKANG1 |
| 38 |                                            | DLOAD  | GOTO     |
| 39 |                                            |        | VECANG2  |
| 40 |                                            |        | COMPMFSN |
| 41 |                                            |        |          |
| 42 | PICKANG1                                   | DLOAD  |          |
| 43 |                                            |        | VECANG1  |
| 44 | COMPMFSN                                   | CALL   |          |
| 45 |                                            |        | DELCOMP  |
| 46 |                                            | AXC,1  | AXC,2    |
| 47 |                                            |        | MIS      |
| 48 |                                            |        | KEL      |
| 49 |                                            | CALL   |          |
| 50 |                                            |        | MXM3     |
| 51 |                                            |        |          |
| 52 |                                            |        |          |
| 53 |                                            |        |          |
| 54 |                                            |        |          |
| 55 |                                            |        |          |
| 56 |                                            |        |          |
| 57 |                                            |        |          |
| 58 |                                            |        |          |
| 59 |                                            |        |          |
| 60 |                                            |        |          |

```
1 # GIMBAL LOCK
2
3 FINDGIMB AXC,1 CALL
4 0 # EXTRACT THE COMMANDED CDU ANGLES FROM
5 DCMTCDU # THIS MATRIX
6 RTB SETPD
7 VISTO2S # CONVERT TO 2:S COMPLEMENT
8 0
9 GOTO
10
11
12 PICKAXIS VLOAD DOT # IF VF X VI = 0, FIND VF . VI
13 28D
14 SCAXIS
15 BMN TLOAD
16 ROT180
17 25D
18 GOTO # IF VF = VI, CDU DESIRED = PRESENT CDU
19 VECQTEMP # PRESENT CDU ANGLES
20
21 BANK 35
22 SETLOC MANUVER1
23 BANK
24 ROT180 VLOAD VXV # IF VF, VI ANTIPARALLEL, 180 DEG ROTATION
25 MIS +6 # IS REQUIRED. Y STABLE MEMBER AXIS IN
26 HIDPHALF # INITIAL S/C AXES.
27 UNIT VXV # FIND Y(SM) X X(I)
28 SCAXIS # FIND UNIT(VI X UNIT(Y(SM) X X(I)))
29 UNIT BOV # I.E. PICK A VECTOR IN THE PLANE OF X(I),
30 PICKX # Y(SM) PERPENDICULAR TO VI
31 STODL COF
32 36D # CHECK MAGNITUDE
33 DSU BMN # OF THIS VECTOR.
34 DPB-14 # IF LESS THAN B-14,
35 PICKX # PICK X-AXIS.
36 VLOAD
37 XROT STODL COF
38 COF
39 HIDPHALF
40 GOTO
41
42 PICKX VLOAD COMPMATX
43 GOTO # PICK THE XAXIS IN THIS CASE
44 HIDPHALF
45 SINGIMLC 2DEC .4285836003 # =SIN(59) $2
46
47 SINVEC1 2DEC .3796356537 # =SIN(49.4) $2
48
49 SINVEC2 2DEC .2462117800 # =SIN(29.5) $2
50
51 VECANG1 2DEC .1388888889 # = 50 DEGREES $360
52
53
54
55
56
57
58
59
60
```

```
VECANG2 2DEC .0972222222 # = 35 DEGREES $360
```

|        |     |       |                             |       |
|--------|-----|-------|-----------------------------|-------|
| 1BITDP | OCT | 0     | # KEEP THIS BEFORE DPB(-14) | ***** |
| DPB-14 | OCT | 00001 |                             |       |
|        | OCT | 00000 |                             |       |



|    |                                                                |        |          |                                        |    |
|----|----------------------------------------------------------------|--------|----------|----------------------------------------|----|
| 1  | # ROUTINE FOR INITIATING AUTOMATIC MANEUVER VIA KEYBOARD (V49) |        |          |                                        | 1  |
| 2  |                                                                |        |          |                                        | 2  |
| 3  |                                                                |        |          |                                        | 3  |
| 4  |                                                                | BANK   | 34       |                                        | 4  |
| 5  |                                                                | SETLOC | R62      |                                        | 5  |
| 6  |                                                                | BANK   |          |                                        | 6  |
| 7  |                                                                | EBANK= | BCDU     |                                        | 7  |
| 8  |                                                                |        |          |                                        | 8  |
| 9  |                                                                | COUNT* | \$\$/R62 |                                        | 9  |
| 10 |                                                                |        |          |                                        | 10 |
| 11 | R62DISP                                                        | EQUALS | R62FLASH |                                        | 11 |
| 12 |                                                                |        |          |                                        | 12 |
| 13 | R62FLASH                                                       | CAF    | V06N22   | # FLASH V06N22 AND                     | 13 |
| 14 |                                                                | TC     | BANKCALL | # ICDU ANGLES                          | 14 |
| 15 |                                                                | CADR   | GOFLASH  |                                        | 15 |
| 16 |                                                                | TCF    | ENDEXT   | # TERMINATE                            | 16 |
| 17 |                                                                | TCF    | GOMOVE   | # PROCEED                              | 17 |
| 18 |                                                                | TCF    | R62FLASH | # ENTER                                | 18 |
| 19 |                                                                |        |          |                                        | 19 |
| 20 |                                                                |        |          | # ASTRONAUT MAY LOAD NEW ICDUS AT THIS | 20 |
| 21 |                                                                |        |          | # POINT                                | 21 |
| 22 | GOMOVE                                                         | TC     | UPFLAG   | # SET FOR 3-AXIS MANEUVER              | 22 |
| 23 |                                                                | ADRES  | 3AXISFLG |                                        | 23 |
| 24 |                                                                |        |          |                                        | 24 |
| 25 |                                                                | TC     | BANKCALL |                                        | 25 |
| 26 |                                                                | CADR   | R60LEM   |                                        | 26 |
| 27 |                                                                | TCF    | ENDEXT   | # END R62                              | 27 |
| 28 |                                                                |        |          |                                        | 28 |
| 29 |                                                                |        |          |                                        | 29 |
| 30 |                                                                |        |          |                                        | 30 |
| 31 |                                                                |        |          |                                        | 31 |
| 32 |                                                                |        |          |                                        | 32 |
| 33 |                                                                |        |          |                                        | 33 |
| 34 |                                                                |        |          |                                        | 34 |
| 35 |                                                                |        |          |                                        | 35 |
| 36 |                                                                |        |          |                                        | 36 |
| 37 |                                                                |        |          |                                        | 37 |
| 38 |                                                                |        |          |                                        | 38 |
| 39 |                                                                |        |          |                                        | 39 |
| 40 |                                                                |        |          |                                        | 40 |
| 41 |                                                                |        |          |                                        | 41 |
| 42 |                                                                |        |          |                                        | 42 |
| 43 |                                                                |        |          |                                        | 43 |
| 44 |                                                                |        |          |                                        | 44 |
| 45 |                                                                |        |          |                                        | 45 |
| 46 |                                                                |        |          |                                        | 46 |
| 47 |                                                                |        |          |                                        | 47 |
| 48 |                                                                |        |          |                                        | 48 |
| 49 |                                                                |        |          |                                        | 49 |
| 50 |                                                                |        |          |                                        | 50 |
| 51 |                                                                |        |          |                                        | 51 |
| 52 |                                                                |        |          |                                        | 52 |
| 53 |                                                                |        |          |                                        | 53 |
| 54 |                                                                |        |          |                                        | 54 |
| 55 |                                                                |        |          |                                        | 55 |
| 56 |                                                                |        |          |                                        | 56 |
| 57 |                                                                |        |          |                                        | 57 |
| 58 |                                                                |        |          |                                        | 58 |
| 59 |                                                                |        |          |                                        | 59 |
| 60 |                                                                |        |          |                                        | 60 |

# S-BAND\_ANTENNA\_FOR\_LM

```
SUBROUTINE NAME: R05 - S-BAND ANTENNA FOR LM
#
MOD0 BY T. JAMES
MOD1 BY P. SHAKIR
#
FUNCTIONAL DESCRIPTION
#
THE S-BAND ANTENNA ROUTINE, R05, COMPUTES AND DISPLAYS THE PITCH AND
YAW ANTENNA GIMBAL ANGLES REQUIRED TO POINT THE LM STEERABLE ANTENNA
TOWARD THE CENTER OF THE EARTH. THIS ROUTINE IS SELECTED BY THE ASTRO-
NAUT VIA DSKY ENTRY DURING COASTING FLIGHT OR WHEN THE LM IS ON THE MOON
SURFACE. THE EARTH OR MOON REFERENCE COORDINATE SYSTEM IS USED DEPENDING
ON WHETHER THE LM IS ABOUT TO ENTER OR HAS ALREADY ENTERED THE MOON
SPHERE OF INFLUENCE, RESPECTIVELY
#
TO CALL SUBROUTINE, ASTRONAUT KEYS IN V 64 E
#
SUBROUTINES CALLED-
R02BOTH
INTPRET
LOADTIME
LEMCONIC
LUNPOS
CDUTRIG
SMNB
BANKCALL
B500FF
ENDOFJOB
BLANKET
#
RETURNS WITH
PITCH ANGLE IN PITCHANG REV. B0
YAW ANGLE IN YAWANG REV. B0
#
ERASABLES USED
PITCHANG
YAWANG
RLM
VAC AREA
#
BANK 41
SETLOC SBAND
BANK
#
EBANK= WHOCARES
COUNT* $$/R05
SBANDANT TC BANKCALL
```

1412THE

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60

1



```
1
2 CADR R02BOTH # CHECK IF IMU IS ON AND ALIGNED
3 TC INTPRET
4 SETPD RTB
5 OD
6 LOADTIME # PICK UP CURRENT TIME
7 STCALL TDEC1 # ADVANCE INTEGRATION TO TIME IN TDEC1
8 LEMCONIC # USING CONIC INTEGRATION
9 SLOAD BHIZ
10 X2 # X2 =0 EARTH SPHERE, X2 =2 MOON SPHERE
11 CONV4
12 VLOAD
13 STODL RATT
14 RLM
15 TAT
16 CONV3 CALL
17 LUNPOS # UNIT POSITION VECTOR FROM EARTH TO MOON
18 VLOAD VXSC
19 VMOON
20 REMDIST # MEAN DISTANCE FROM EARTH TO MOON
21 VSL1 VAD
22 RLM
23 GOTO CONV5
24 CONV4 VLOAD
25 CONV5 SETPD # UE = -UNIT(RATT) EARTH SPHERE
26 UNIT # UE = -UNIT((REM)(UEM) + RL) MOON SPHERE
27 OD # SET PL POINTER TO 0
28 VCOMP CALL
29 CDUTRIG # COMPUTE SINES AND COSINES OF CDU ANGLES
30 MXV VSL1 # TRANSFORM REF. COORDINATE SYSTEM TO
31 REFSMMAT # STABLE MEMBER B-1 X B-1 X B+1 = B-1
32 PUSH DLOAD # 8D
33 HI6ZEROS
34 STORE PITCHANG
35 STOVL YAWANG # ZERO OUT ANGLES
36 CALL
37 *SMNB*
38 STODL RLM # PRE-MULTIPLY RLM BY (NBSA) MATRIX(B0)
39 RLM +2
40 PUSH DSU
41 RLM
42 DMP
43 10VSQRT2
44 STODL RLM +2
45 DAD DMP
46 RLM
47 10VSQRT2
48 STOVL RLM # R B-1
49 RLM
50 UNIT PDVL
```

|          |        |          |                                    |
|----------|--------|----------|------------------------------------|
|          |        | RLM      |                                    |
|          | VPROJ  | VSL2     | # PROJECTION OF R ONTO LM XZ PLANE |
|          |        | HIUNITY  |                                    |
|          | BVSU   | BOV      | # CLEAR OVERFLOW INDICATOR IF ON   |
|          |        | RLM      |                                    |
| COVCNV   | UNIT   | COVCNV   |                                    |
|          |        | BOV      | # EXIT ON OVERFLOW                 |
|          |        | SBANDEX  |                                    |
|          | PUSH   | VXV      | # URP VECTOR B-1                   |
|          |        | HIUNITZ  |                                    |
|          | VSL1   | VCOMP    | # UZ X URP = -(URP X UZ)           |
|          | STORE  | RLM      | # X VEC B-1                        |
|          | DOT    | PDVL     | # SGN(X.UY) UNSCALED               |
|          |        | HIUNITY  |                                    |
|          |        | RLM      |                                    |
|          | ABVAL  | SIGN     |                                    |
|          | ASIN   |          | # ASIN((SGN(X.UY))ABV(X)) REV B0   |
|          | STOVL  | PITCHANG |                                    |
|          |        | URP      |                                    |
|          | DOT    | BPL      |                                    |
|          |        | HIUNITZ  |                                    |
|          |        | NOADJUST | # YES, -90 TO +90                  |
|          | DLOAD  | DSU      |                                    |
|          |        | HIDPHALF |                                    |
|          |        | PITCHANG |                                    |
|          | STORE  | PITCHANG |                                    |
| NOADJUST | VLOAD  | VXV      |                                    |
|          |        | UR       | # Z = (UR X URP)                   |
|          |        | URP      |                                    |
|          | VSL1   |          |                                    |
|          | STODL  | RLM      | # Z VEC B-1                        |
|          |        | PITCHANG |                                    |
|          | SIN    | VXSC     |                                    |
|          |        | HIUNITZ  |                                    |
|          | PDDL   | COS      |                                    |
|          |        | PITCHANG |                                    |
|          | VXSC   | VSU      |                                    |
|          |        | HIUNITX  | # (UX COS ALPHA) - (UZ SIN ALPHA)  |
|          | DOT    | PDVL     | # YAW.Z                            |
|          |        | RLM      |                                    |
|          |        | RLM      |                                    |
|          | ABVAL  | SIGN     |                                    |
|          | ASIN   |          |                                    |
|          | STORE  | YAWANG   |                                    |
| SBANDEX  | EXIT   |          |                                    |
|          | CA     | EXTVBACT |                                    |
|          | MASK   | BIT5     | # IS BIT5 STILL ON                 |
|          | EXTEND |          |                                    |
|          | BZF    | ENDEXT   | # NO                               |
|          | CAF    | PRI05    |                                    |

```
*** END OF LNYAIDE .001 ***
```

```
1
2 BANK 25
3 SETLOC RRLEADIN
4 BANK
5
6 EBANK= RSTACK
7
8 # RADAR SAMPLING LOOP.
9
10 RADSAMP COUNT* $$/RLEAD
11 CCS RSAMPDT # TIMES NORMAL ONCE-PER-SECOND SAMPLING.
12 TCF +2
13 TCF TASKOVER # +0 INSERTED MANUALLY TERMINATES TEST.
14
15 TC WAITLIST
16 EBANK= RSTACK
17 2CADR RADSAMP
18
19 CAF PRI025
20 TC NOVAC
21 EBANK= RSTACK
22 2CADR DORSAMP
23
24 CAF BIT14 # FOR CYCLIC SAMPLING, RTSTDEX =
25 EXTEND # RTSTLOC/2 + RTSTBASE
26 MP RTSTLOC
27 AD RTSTBASE # 0 FOR RR, 2 FOR LR.
28 TS RTSTDEX
29 TCF TASKOVER
30
31 # DO THE ACTUAL RADAR SAMPLE.
32
33 DORSAMP TC VARADAR # SELECTS VARIABLE RADAR CHANNEL.
34 TC BANKCALL
35 CADR RADSTALL
36
37 INCR RFAILCNT # ADVANCE FAIL COUNTER BUT ACCEPT BAD DATA
38
39 DORSAMP2 INHINT
40 CA FLAGWRD5 # DON'T UPDATE RSTACK IF IN R77.
41 MASK R77FLBIT
42 CCS A
43 TCF +4
44
45 DXCH SAMPLSUM
46 INDEX RTSTLOC
47 DXCH RSTACK
48
49 CS RTSTLOC # CYCLE RTSTLOC.
50 AD RTSTMAX
51 EXTEND
```



|    |                                                        |       |          |                                            |    |
|----|--------------------------------------------------------|-------|----------|--------------------------------------------|----|
| 1  |                                                        |       |          |                                            | 1  |
| 2  |                                                        | BZF   | +3       |                                            | 2  |
| 3  |                                                        | CA    | RTSTLOC  |                                            | 3  |
| 4  |                                                        | AD    | TWO      | # STORAGE IS DP                            | 4  |
| 5  |                                                        | TS    | RTSTLOC  |                                            | 5  |
| 6  |                                                        | TCF   | ENDOFJOB | # CONTINUOUS SAMPLING AND 2N TRIES - GONE. | 6  |
| 7  |                                                        |       |          |                                            | 7  |
| 8  | # VARIABLE RADAR DATA CALLER FOR ONE MEASUREMENT ONLY. |       |          |                                            | 8  |
| 9  |                                                        |       |          |                                            | 9  |
| 10 | VARADAR                                                | CAF   | ONE      | # WILL BE SENT TO RADAR ROUTINE IN A BY    | 10 |
| 11 |                                                        | TS    | BUF2     | # SWCALL.                                  | 11 |
| 12 |                                                        | INDEX | RTSTDEX  |                                            | 12 |
| 13 |                                                        | CAF   | RDRLOCS  |                                            | 13 |
| 14 |                                                        | TCF   | SWCALL   | # NOT TOUCHING Q.                          | 14 |
| 15 |                                                        |       |          |                                            | 15 |
| 16 | RDRLOCS                                                | CADR  | RRRANGE  | # =0                                       | 16 |
| 17 |                                                        | CADR  | RRRDOT   | # =1                                       | 17 |
| 18 |                                                        | CADR  | LRVELX   | # =2                                       | 18 |
| 19 |                                                        | CADR  | LRVELY   | # =3                                       | 19 |
| 20 |                                                        | CADR  | LRVELZ   | # =4                                       | 20 |
| 21 |                                                        | CADR  | LRALT    | # =5                                       | 21 |
| 22 |                                                        |       |          |                                            | 22 |
| 23 |                                                        |       |          |                                            | 23 |
| 24 |                                                        |       |          |                                            | 24 |
| 25 |                                                        |       |          |                                            | 25 |
| 26 |                                                        |       |          |                                            | 26 |
| 27 |                                                        |       |          |                                            | 27 |
| 28 |                                                        |       |          |                                            | 28 |
| 29 |                                                        |       |          |                                            | 29 |
| 30 |                                                        |       |          |                                            | 30 |
| 31 |                                                        |       |          |                                            | 31 |
| 32 |                                                        |       |          |                                            | 32 |
| 33 |                                                        |       |          |                                            | 33 |
| 34 |                                                        |       |          |                                            | 34 |
| 35 |                                                        |       |          |                                            | 35 |
| 36 |                                                        |       |          |                                            | 36 |
| 37 |                                                        |       |          |                                            | 37 |
| 38 |                                                        |       |          |                                            | 38 |
| 39 |                                                        |       |          |                                            | 39 |
| 40 |                                                        |       |          |                                            | 40 |
| 41 |                                                        |       |          |                                            | 41 |
| 42 |                                                        |       |          |                                            | 42 |
| 43 |                                                        |       |          |                                            | 43 |
| 44 |                                                        |       |          |                                            | 44 |
| 45 |                                                        |       |          |                                            | 45 |
| 46 |                                                        |       |          |                                            | 46 |
| 47 |                                                        |       |          |                                            | 47 |
| 48 |                                                        |       |          |                                            | 48 |
| 49 |                                                        |       |          |                                            | 49 |
| 50 |                                                        |       |          |                                            | 50 |
| 51 |                                                        |       |          |                                            | 51 |
| 52 |                                                        |       |          |                                            | 52 |
| 53 |                                                        |       |          |                                            | 53 |
| 54 |                                                        |       |          |                                            | 54 |
| 55 |                                                        |       |          |                                            | 55 |
| 56 |                                                        |       |          |                                            | 56 |
| 57 |                                                        |       |          |                                            | 57 |
| 58 |                                                        |       |          |                                            | 58 |
| 59 |                                                        |       |          |                                            | 59 |
| 60 |                                                        |       |          |                                            | 60 |

```
1 # RENDEZVOUS NAVIGATION PROGRAM 20
2 #
3 # PROGRAM DESCRIPTION
4 #
5 # MOD NO -- 2
6 # BY P. VOLANTE
7 #
8 # FUNCTIONAL DESCRIPTION
9 #
10 # THE PURPOSE OF THIS PROGRAM IS TO CONTROL THE RENDEZVOUS RADAR FROM
11 # STARTUP THROUGH ACQUISITION AND LOCKON TO THE CSM AND TO UPDATE EITHER
12 # THE LM OR CSM STATE VECTOR (AS SPECIFIED BY THE ASTRONAUT BY DSKY ENTRY)
13 # ON THE BASIS OF THE RR TRACKING DATA.
14 #
15 # CALLING SEQUENCE --
16 #
17 # ASTRONAUT REQUEST THROUGH DSKY V37E20E
18 #
19 # SUBROUTINES CALLED
20 #
21 # R02BOTH (IMU STATUS CHECK) FLAGUP
22 # GOFLASH (PINBALL-DISPLAY) FLAGDOWN
23 # R23LEM (MANUAL ACQUISITION) BANKCALL
24 # LS201 (LOS DETERMINATION) TASKOVER
25 # LS202 (RANGE LIMIT TEST)
26 # R61LEM (PREFERRED TRACKING ATTITUDE)
27 # R21LEM (RR DESIGNATE) ENDOFJOB
28 # R22LEM (DATA READ) GOPERF1
29 # R31LEM (RENDEZVOUS PARAMETER DISPLAY)
30 # PRIOLARM (PRIORITY DISPLAY)
31 #
32 # NORMAL EXIT MODES --
33 #
34 # P20 MAY BE TERMINATED IN TWO WAYS -- ASTRONAUT SELECTION OF IDLING
35 # PROGRAM (P00) BY KEYING V37E00E OR BY KEYING IN V56E
36 #
37 # ALARM OR ABORT EXIT MODES --
38 #
39 # RANGE GREATER THAN 400 NM DISPLAY
40 #
41 # OUTPUT
42 #
43 # TRKMKCNT = NO OF RENDEZVOUS TRACKING MARKS TAKEN (COUNTER)
44 #
45 # ERASABLE INITIALIZATION REQUIRED
46 #
47 # FLAGS SET + RESET
48 #
49 # SRCHOPT, RNDVZFLG, ACMODFLG, VEHUPFLG, UPDATFLG, TRACKFLG
50 #
51 # DEBRIS
52 #
53 # CENTRALS -- A,Q,L
54 #
55 # SBANK= LOWSUPER # FOR LOW 2CADR'S.
56 #
57 # BANK 33
58 # SETLOC P20S
59 # BANK
```

|    |        |        |          |                             |    |
|----|--------|--------|----------|-----------------------------|----|
| 1  |        | EBANK= | LOSCOUNT |                             | 1  |
| 2  |        | COUNT* | \$/P20   |                             | 2  |
| 3  | PROG22 | =      | PROG20   |                             | 3  |
| 4  | PROG20 | TC     | 2PHSCHNG |                             | 4  |
| 5  |        | OCT    | 4        |                             | 5  |
| 6  |        | OCT    | 05022    |                             | 6  |
| 7  |        | OCT    | 26000    | # PRIORITY 26               | 7  |
| 8  |        | TC     | LUNSFCHK | # CHECK IF ON LUNAR SURFACE | 8  |
| 9  |        |        |          |                             | 9  |
| 10 |        |        |          |                             | 10 |
| 11 |        |        |          |                             | 11 |
| 12 |        |        |          |                             | 12 |
| 13 |        |        |          |                             | 13 |
| 14 |        |        |          |                             | 14 |
| 15 |        |        |          |                             | 15 |
| 16 |        |        |          |                             | 16 |
| 17 |        |        |          |                             | 17 |
| 18 |        |        |          |                             | 18 |
| 19 |        |        |          |                             | 19 |
| 20 |        |        |          |                             | 20 |
| 21 |        |        |          |                             | 21 |
| 22 |        |        |          |                             | 22 |
| 23 |        |        |          |                             | 23 |
| 24 |        |        |          |                             | 24 |
| 25 |        |        |          |                             | 25 |
| 26 |        |        |          |                             | 26 |
| 27 |        |        |          |                             | 27 |
| 28 |        |        |          |                             | 28 |
| 29 |        |        |          |                             | 29 |
| 30 |        |        |          |                             | 30 |
| 31 |        |        |          |                             | 31 |
| 32 |        |        |          |                             | 32 |
| 33 |        |        |          |                             | 33 |
| 34 |        |        |          |                             | 34 |
| 35 |        |        |          |                             | 35 |
| 36 |        |        |          |                             | 36 |
| 37 |        |        |          |                             | 37 |
| 38 |        |        |          |                             | 38 |
| 39 |        |        |          |                             | 39 |
| 40 |        |        |          |                             | 40 |
| 41 |        |        |          |                             | 41 |
| 42 |        |        |          |                             | 42 |
| 43 |        |        |          |                             | 43 |
| 44 |        |        |          |                             | 44 |
| 45 |        |        |          |                             | 45 |
| 46 |        |        |          |                             | 46 |
| 47 |        |        |          |                             | 47 |
| 48 |        |        |          |                             | 48 |
| 49 |        |        |          |                             | 49 |
| 50 |        |        |          |                             | 50 |
| 51 |        |        |          |                             | 51 |
| 52 |        |        |          |                             | 52 |
| 53 |        |        |          |                             | 53 |
| 54 |        |        |          |                             | 54 |
| 55 |        |        |          |                             | 55 |
| 56 |        |        |          |                             | 56 |
| 57 |        |        |          |                             | 57 |
| 58 |        |        |          |                             | 58 |
| 59 |        |        |          |                             | 59 |
| 60 |        |        |          |                             | 60 |

|         |        |            |                                        |
|---------|--------|------------|----------------------------------------|
|         | TC     | ORBCHGO    | # YES                                  |
|         | TC     | PROG20A -2 | # NO -- CONTINUE WITH P20              |
| ORBCHGO | TC     | UPFLAG     | # SET VEHUPFLG -- CSM STATE            |
|         | ADRES  | VEHUPFLG   | # VECTOR TO BE UPDATED                 |
|         | CAF    | ONE        | # SET R2 FOR OPTION CSM WILL NOT       |
|         | TS     | OPTION2    | # CHANGE PRESENT ORBIT                 |
|         | CAF    | OCT00012   |                                        |
|         | TC     | BANKCALL   | # DISPLAY ASSUMED CSM ORBIT OPTION     |
|         | CADR   | GOPERF4    |                                        |
|         | TC     | GOTOP00H   | # TERMINATE                            |
|         | TC     | ORBCHG1    | # PROCEED VALUE OF ASSUMED OPTION OK   |
| ORBCHG1 | TC     | -5         | # R2 LOADED THRU DSKY                  |
|         | CS     | P22ONE     |                                        |
|         | AD     | OPTION2    |                                        |
|         | EXTEND |            |                                        |
|         | BZF    | PROG20A    |                                        |
|         | CAF    | V06N33*    |                                        |
|         | TC     | BANKCALL   | # FLASH VERB-NOUN TO REQUEST ESTIMATED |
|         | CADR   | GOFLASH    | # TIME OF LAUNCH                       |
|         | TC     | GOTOP00H   | # TERMINATE                            |
|         | TC     | ORBCHG2    | # PROCEED VALUES OK                    |
|         | TC     | -5         | # TIME LOADED THRU DSKY                |
| ORBCHG2 | TC     | INTPRET    |                                        |
|         | GOTO   |            |                                        |
|         |        | ORBCHG3    |                                        |
|         | BANK   | 32         |                                        |
|         | SETLOC | P20S4      |                                        |
|         | BANK   |            |                                        |
|         | COUNT* | \$\$/P20   |                                        |
| ORBCHG3 | CALL   |            |                                        |
|         |        | INTSTALL   |                                        |
|         | DLOAD  |            |                                        |
|         |        | TIG        |                                        |
|         | STORE  | LNCHTM     |                                        |
|         | STORE  | TDEC1      | # ESTIMATED LAUNCH TIME                |
|         | CLEAR  | CLEAR      |                                        |
|         |        | VINTFLAG   | # LM INTEGRATION                       |
|         |        | INTYPFLG   | # PRECISION -- ENCKE                   |
|         | CLEAR  | CLEAR      |                                        |
|         |        | DIM0FLAG   | # NO W-MATRIX                          |
|         |        | D6OR9FLG   |                                        |
|         | CALL   |            |                                        |
|         |        | INTEGRV    | # PLANETARY INERTIAL ORIENTATION       |
|         | CALL   |            |                                        |
|         |        | GRP2PC     |                                        |
|         | VLOAD  |            |                                        |
|         |        | RATT1      |                                        |
|         | STODL  | RSUBL      | # SAVE LM POSITION                     |
|         |        | TAT        |                                        |



|    |         |        |          |                                       |
|----|---------|--------|----------|---------------------------------------|
| 1  |         | STCALL | TDEC1    |                                       |
| 2  |         |        | INTSTALL |                                       |
| 3  |         |        |          |                                       |
| 4  |         | SET    | CLEAR    |                                       |
| 5  |         |        | VINTFLAG | # CSM INTEGRATION                     |
| 6  |         |        | INTYPFLG |                                       |
| 7  |         | CLEAR  | BOFF     |                                       |
| 8  |         |        | DIM0FLAG |                                       |
| 9  |         |        | RENDWFLG | # W MATRIX VALID                      |
| 10 |         |        | NOWMATX  | # NO                                  |
| 11 |         | SET    | SET      | # YES -- SET FOR W MATRIX             |
| 12 |         |        | DIM0FLAG |                                       |
| 13 |         |        | D6OR9FLG |                                       |
| 14 | NOWMATX | CALL   |          |                                       |
| 15 |         |        | INTEGRV  | # CSM INTEGRATION                     |
| 16 |         | CALL   |          |                                       |
| 17 |         |        | GRP2PC   |                                       |
| 18 |         | VLOAD  |          |                                       |
| 19 |         |        |          |                                       |
| 20 |         | STOVL  | VATT1    |                                       |
| 21 |         |        | VSUBC    | # SAVE CSM POSITION                   |
| 22 |         |        | RATT1    |                                       |
| 23 |         | STORE  | RSUBC    | # SAVE CSM POSITION                   |
| 24 |         | VXV    | UNIT     | # COMPUTE NORMAL TO CSM ORBITAL PLANE |
| 25 |         |        | VSUBC    | # NSUB1=UNIT(R(CM) CROSS V(CM))       |
| 26 |         | STOVL  | 20D      | # SAVE NSUB1                          |
| 27 |         |        | RSUBL    | # COMPUTE ESTIMATED ORBITAL           |
| 28 |         | VXV    | UNIT     | # PLANE CHANGE                        |
| 29 |         |        | 20D      | # UCSM = UNIT(R(LM) CROSS NSUB1)      |
| 30 |         | STOVL  | UCSM     |                                       |
| 31 |         |        | RSUBC    | # COMPUTE ANGLE BETWEEN UCSM          |
| 32 |         | UNIT   | DOT      | # AND RSUBC                           |
| 33 |         |        | UCSM     | # COS A = UCSM DOT UNIT (R(CM))       |
| 34 |         | SL1    |          |                                       |
| 35 |         | STORE  | CSTH     | # SAVE DOE TIME-THETA SUBROUTINE      |
| 36 |         | DSQ    | BDSU     | # COMPUTE SINE A                      |
| 37 |         |        | ONEB-2   |                                       |
| 38 |         | SQRT   |          |                                       |
| 39 |         | STOVL  | SNTH     | # SAVE FOR TIME-THETA SUBROUTINE      |
| 40 |         |        | RSUBC    | # POSITION OF CSM AT EST. LAUNCH      |
| 41 |         | STOVL  | RVEC     | # TIME FOR TIME-THETHA B-27           |
| 42 |         |        | VSUBC    | # VELOCITY OF CSM AT EST. LAUNCH.     |
| 43 |         | VCOMP  |          |                                       |
| 44 |         | STORE  | VVEC     | # TIME FOR TIME THETA B-5             |
| 45 |         | CLEAR  | CALL     |                                       |
| 46 |         |        | RVSW     |                                       |
| 47 |         |        | TIMETHET |                                       |
| 48 |         | VCOMP  |          |                                       |
| 49 |         | STORE  | NEWVEL   | # TERMINAL VELOCITY OF CSM            |
| 50 |         | DLOAD  |          |                                       |
| 51 |         |        | T        |                                       |
| 52 |         | STOVL  | TRANSTM  | # TRANSFER TIME                       |

|    |         |          |   |                                      |
|----|---------|----------|---|--------------------------------------|
| 1  |         |          |   |                                      |
| 2  |         | NEWVEL   |   |                                      |
| 3  | ABVAL   |          |   |                                      |
| 4  | STOVL   | 20D      |   |                                      |
| 5  |         | OD       |   |                                      |
| 6  | STORE   | NEWPOS   | # | TERMINAL POSITION OF CSM             |
| 7  | VXV     | UNIT     | # | COMPUTE NORMAL TO SCM ORBITAL PLANE  |
| 8  |         | RSUBL    | # | NSUB2 = UNIT(NEWPOS CROSS R(LM))     |
| 9  | VXV     | UNIT     | # | ROTATE TERMINAL VEL INTO DESIRED     |
| 10 |         | NEWPOS   | # | ORBITAL PLANE                        |
| 11 | VXSC    | VSL1     | # | VSUBC = ABVAL(NEWVEL) \$ UNIT (NSUB2 |
| 12 |         | 20D      |   |                                      |
| 13 | STCALL  | NCSMVEL  | # | NEW CSM VELOCITY                     |
| 14 |         | GRP2PC   |   |                                      |
| 15 | CALL    |          |   |                                      |
| 16 |         | INTSTALL |   |                                      |
| 17 | DLOAD   | BDSU     |   |                                      |
| 18 |         | TRANSTM  | # | LAUNCH TIME -- TRANSFER TIME         |
| 19 |         | LNCHTM   |   |                                      |
| 20 | STOVL   | TET      |   |                                      |
| 21 |         | NEWPOS   |   |                                      |
| 22 | STORE   | RCV      |   |                                      |
| 23 | STOVL   | RRECT    |   |                                      |
| 24 |         | NCSMVEL  |   |                                      |
| 25 | STCALL  | VRECT    |   |                                      |
| 26 |         | MINIRECT |   |                                      |
| 27 | AXT,2   | CALL     |   |                                      |
| 28 |         | 2        |   |                                      |
| 29 |         | ATOPCSM  |   |                                      |
| 30 | CALL    |          |   |                                      |
| 31 |         | INTWAKEO |   |                                      |
| 32 | EXIT    |          |   |                                      |
| 33 | TC      | BANKCALL |   |                                      |
| 34 | CADR    | PROG20A  |   |                                      |
| 35 |         |          |   |                                      |
| 36 | BANK    | 24       |   |                                      |
| 37 | SETLOC  | P20S     |   |                                      |
| 38 | BANK    |          |   |                                      |
| 39 | COUNT*  | \$\$/P20 |   |                                      |
| 40 |         |          |   |                                      |
| 41 | TC      | DOWNFLAG | # | RESET VEHUPFLG -- LM STATE VECTOR    |
| 42 | ADRES   | VEHUPFLG | # | TO BE UPDATED                        |
| 43 | PROG20A | BANKCALL |   |                                      |
| 44 | CADR    | R02BOTH  |   |                                      |
| 45 | TC      | UPFLAG   |   |                                      |
| 46 | ADRES   | UPDATFLG | # | SET UPDATE FLAG                      |
| 47 | TC      | UPFLAG   |   |                                      |
| 48 | ADRES   | TRACKFLG | # | SET TRACK FLAG                       |
| 49 | TC      | UPFLAG   |   |                                      |
| 50 | ADRES   | RNDVZFLG | # | SET RENDEZVOUS FLAG                  |
| 51 | TC      | DOWNFLAG |   |                                      |
| 52 | ADRES   | SRCHOPTN | # | INSURE SEARCH OPTION OFF             |
| 53 |         |          |   |                                      |
| 54 |         |          |   |                                      |
| 55 |         |          |   |                                      |
| 56 |         |          |   |                                      |
| 57 |         |          |   |                                      |
| 58 |         |          |   |                                      |
| 59 |         |          |   |                                      |
| 60 |         |          |   |                                      |

|          |        |          |                                         |
|----------|--------|----------|-----------------------------------------|
|          | TC     | DOWNFLAG | # ALSO MANUAL ACQUISITION FLAG RESET    |
|          | ADRES  | ACMODFLG |                                         |
|          | TC     | DOWNFLAG | # TURN OFF R04FLAG TO ENSURE GETTING    |
|          | ADRES  | R04FLAG  | # ALARM 521 IF CAN'T READ RADAR         |
|          | TC     | DOWNFLAG | # ENSURE R25 GIMBAL MONITOR IS ENABLED  |
|          | ADRES  | NORRMON  | # (RESET NORRMON FLAG)                  |
|          | TC     | DOWNFLAG | # RESET LOS BEING COMPUTED FLAG         |
|          | ADRES  | LOSCMFLG |                                         |
| P20LEM1  | TC     | CLRADMOD |                                         |
|          | TC     | PHASCHNG |                                         |
|          | OCT    | 04022    |                                         |
|          | CAF    | ZERO     | # ZERO MARK COUNTER                     |
|          | TS     | MARKCTR  |                                         |
|          | TC     | INTPRET  | # LOS DETERMINATION ROUTINE             |
|          | RTB    |          |                                         |
|          | STCALL | LOADTIME |                                         |
|          |        | TDEC1    |                                         |
|          |        | LPS20.1  |                                         |
|          | CALL   |          |                                         |
|          |        | LPS20.2  | # TEST RANGE R/UTINE                    |
|          | EXIT   |          |                                         |
|          | INDEX  | MPAC     |                                         |
|          | TC     | +1       |                                         |
| 526ALARM | TC     | P20LEMA  | # NORMAL RETURN WITHIN 400 N M          |
|          | CAF    | ALRM526  | # ERROR EXIT -- RANGE > 400 N. MI.      |
|          | TC     | BANKCALL |                                         |
|          | CADR   | PRIOLARM |                                         |
|          | TC     | GOTOV56  | # TERMINATE EXITS P20 VIA V56 CODING    |
|          | TC     | -4       | # PROC (ILLEGAL)                        |
|          | TC     | P20LEM1  | # ENTER RECYCLE                         |
|          | TC     | ENDOFJOB |                                         |
| P20LEMA  | TC     | PHASCHNG |                                         |
|          | OCT    | 04022    |                                         |
|          | TC     | LUNSFCHK | # CHECK LUNAR SURFACE FLAG (P22 FLAG)   |
|          | TC     | P20LEMB  |                                         |
|          | TC     | BANKCALL |                                         |
|          | CADR   | R61LEM   | # PREFERRED TRACKING ATTITUDE ROUTINE   |
| P20LEMB  | TC     | PHASCHNG |                                         |
|          | OCT    | 05022    | # RESTART AT PRIORITY 10 TO ALLOW V37   |
|          | OCT    | 10000    | # REQUESTED PROGRAM TO RUN FIRST        |
|          | CAF    | PRI026   | # RESTORE PRIORITY 26                   |
|          | TC     | PRI0CHNG |                                         |
|          | CA     | FLAGWRD1 | # IS THE TRACK FLAG SET                 |
|          | MASK   | TRACKBIT |                                         |
|          | EXTEND |          |                                         |
|          | BZF    | P20LEMWT | # BRANCH -- NO -- WAIT FOR IT TO BE SET |
| P20LEMB7 | CAF    | BIT2     | # IS RR AUTO MODE DISCRETE PRESENT      |
|          | EXTEND |          |                                         |

|          |                |          |                                                 |
|----------|----------------|----------|-------------------------------------------------|
|          | RAND<br>EXTEND | CHAN33   |                                                 |
|          | BZF            | P20LEMB3 | # YES -- DO AUTOMATIC ACQUISITION (R21)         |
| P20LEMB5 | CS             | OCT24    | # RADAR NOT IN AUTO CHECK IF                    |
|          | AD<br>EXTEND   | MODREG   | # MAJOR MODE IS 20                              |
|          | BZF            | P20LEMB6 | # BRANCH -- YES -- OKAY TO DO PLEASE PERFORM    |
|          | AD<br>EXTEND   | NEG2     | # ALSO CHECK FOR P22                            |
|          | BZF            | P20LEMB6 | # BRANCH -- YES OK TO DO PLEASE PERFORM         |
|          | CAF            | ALRM514  | # TRACK FLAG SET -- FLASH PRIORITY ALARM 514 -- |
|          | TC             | BANKCALL | # RADAR GOES OUT OF AUTO MODE WHILE IN USE      |
|          | CADR           | PRIOLARM |                                                 |
|          | TC             | GOTOV56  | # TERMINATE EXITS VIA V56                       |
|          | TC             | P20LEMB  | # PROCEED AND ENTER BOTH GO BACK                |
|          | TC             | P20LEMB  | # TO CHECK AUTO MODE AGAIN                      |
| P20LEMB6 | TC             | ENDOFJOB |                                                 |
|          | CAF            | OCT201   | # REQUEST RR AUTO MODE SELECTION                |
|          | TC             | BANKCALL |                                                 |
|          | CADR           | GOPERF1  |                                                 |
|          | TC             | GOTOV56  | # TERMINATE EXITS P20 VIA V56 CODING            |
|          | TC             | P20LEMB  | # PROCEED CHECKS AUTO MODE DISCRETE AGAIN       |
|          | TC             | LUNSFCHK | # ENTER INDICATES MANUAL ACQUISITION (R23)      |
|          | TC             | P20LEMB2 | # YES -- R23 NOT ALLOWED -- TURN ON OPR ERROR   |
|          | TC             | R23LEM   | # NO -- DO MANUAL ACQUISITION                   |
| P20LEMB1 | TC             | UPFLAG   | # RETURN FROM R23 -- LOCKON ACHIEVED            |
|          | ADRES          | ACMODFLG | # SET MANUAL FLAG AND GO BACK TO CHECK          |
|          | TC             | P20LEMB  | # RR AUTO MODE                                  |
| P20LEMB2 | TC             | FALTON   | # TURNS ON OPERATOR ERROR LIGHT ON DSKY         |
|          | TC             | P20LEMB  | # AND GOES BACK TO CHECK AUTO MODE              |
| P20LEMB3 | CS             | RADMODES | # ARE RR CDUS BEING ZEROED                      |
|          | MASK<br>EXTEND | RCDUOBIT |                                                 |
|          | BZF            | P20LEMB4 | # BRANCH -- YES -- WAIT                         |
|          | CAF            | BIT13-14 | # IS SEARCH OR MANUAL ACQUISITION FLAG SET      |
|          | MASK           | FLAGWRD2 |                                                 |
|          | EXTEND         |          |                                                 |
|          | BZF            | P20LEMC3 | # ZERO MEANS AUTOMATIC RR ACQUISITION           |
|          | TC             | DOWNFLAG | # RESET TO AUTO MODE                            |
|          | ADRES          | SRCHOPTN |                                                 |

|    |          |        |          |                                                  |
|----|----------|--------|----------|--------------------------------------------------|
| 1  |          |        |          |                                                  |
| 2  |          | TC     | DOWNFLAG |                                                  |
| 3  |          | ADRES  | ACMODFLG |                                                  |
| 4  |          | TC     | P20LEMWT | # WAIT 2.5 SECONDS THEN GO TO RR DATA READ       |
| 5  |          |        |          |                                                  |
| 6  | P20LEMB4 | CAF    | 250DEC   |                                                  |
| 7  |          | TC     | BANKCALL | # WAIT 2.5 SECONDS WHILE RR CDUS ARE BEING       |
| 8  |          | CADR   | DELAYJOB | # ZEROED -- THEN GO BACK AND CHECK AGAIN         |
| 9  |          | TC     | P20LEMB3 |                                                  |
| 10 |          |        |          |                                                  |
| 11 | P20LEMC3 | TC     | INTPRET  |                                                  |
| 12 |          | RTB    |          |                                                  |
| 13 |          |        | LOADTIME |                                                  |
| 14 |          | STCALL | TDEC1    |                                                  |
| 15 |          |        | UPPSV    |                                                  |
| 16 | P20LEMC4 | EXIT   |          |                                                  |
| 17 | P20LEMC  | TC     | PHASCHNG |                                                  |
| 18 |          | OCT    | 04022    |                                                  |
| 19 |          | CAE    | FLAGWRDO | # IS THE RENDEZVOUS FLAG SET                     |
| 20 |          | MASK   | RNDVZBIT |                                                  |
| 21 |          | EXTEND |          |                                                  |
| 22 |          | BZF    | ENDOFJOB | # NO -- EXIT P20                                 |
| 23 |          | CAE    | FLAGWRD1 | # IS TRACK FLAG SET (BIT 5 FLAGWORD 1)           |
| 24 |          | MASK   | TRACKBIT |                                                  |
| 25 |          | EXTEND |          |                                                  |
| 26 | P20LEMF  | BZF    | P20LEMD  | # BRANCH -- TRACK FLAG NOT ON -- WAIT 15 SECONDS |
| 27 |          | TC     | R21LEM   |                                                  |
| 28 |          |        |          |                                                  |
| 29 | P20LEMWT | CAF    | 250DEC   |                                                  |
| 30 |          | TC     | TWIDDLE  | # USE INSTEAD OF WAITLIST SINCE SAME BANK        |
| 31 |          | ADRES  | P20LEMC1 | # WAIT 2.5 SECONDS                               |
| 32 |          | CAE    | FLAGWRD1 | # IS TRACK FLAG SET                              |
| 33 |          | MASK   | TRACKBIT |                                                  |
| 34 |          | EXTEND |          |                                                  |
| 35 | P20LMWT1 | BZF    | ENDOFJOB | # NO -- EXIT WITHOUT DOING 2.7 PHASE CHANGE      |
| 36 |          | TC     | PHASCHNG |                                                  |
| 37 |          | OCT    | 40072    |                                                  |
| 38 |          | TC     | ENDOFJOB |                                                  |
| 39 |          |        |          |                                                  |
| 40 | P20LEMC1 | CAE    | FLAGWRDO | # IS RENDEZVOUS FLAG SET                         |
| 41 |          | MASK   | RNDVZBIT |                                                  |
| 42 |          | EXTEND |          |                                                  |
| 43 |          | BZF    | TASKOVER | # NO -- EXIT P20/R22                             |
| 44 |          | CAE    | FLAGWRD1 | # IS TRACK FLAG SET                              |
| 45 |          | MASK   | TRACKBIT |                                                  |
| 46 |          | EXTEND |          |                                                  |
| 47 |          | BZF    | P20LEMC2 | # NO -- DON'T SCHEDULE R22 JOB                   |
| 48 |          |        |          |                                                  |
| 49 |          |        |          |                                                  |
| 50 |          |        |          |                                                  |
| 51 |          |        |          |                                                  |
| 52 |          |        |          |                                                  |
| 53 |          |        |          |                                                  |
| 54 |          |        |          |                                                  |
| 55 |          |        |          |                                                  |
| 56 |          |        |          |                                                  |
| 57 |          |        |          |                                                  |
| 58 |          |        |          |                                                  |
| 59 |          |        |          |                                                  |
| 60 |          |        |          |                                                  |

|    |          |        |          |                                          |  |
|----|----------|--------|----------|------------------------------------------|--|
| 1  |          |        |          |                                          |  |
| 2  |          | CAF    | PRI026   | # YES -- SCHEDULE R22 JOB (RR DATA READ) |  |
| 3  |          | TC     | FINDVAC  |                                          |  |
| 4  |          | EBANK= | LOSCOUNT |                                          |  |
| 5  |          | 2CADR  | R22LEM42 |                                          |  |
| 6  |          |        |          |                                          |  |
| 7  |          | TC     | TASKOVER |                                          |  |
| 8  |          |        |          |                                          |  |
| 9  | P20LEMC2 | TC     | FIXDELAY | # TRACK FLAG NOT SET, WAIT 15 SECONDS    |  |
| 10 |          | DEC    | 1500     | # AND CHECK AGAIN                        |  |
| 11 |          |        |          |                                          |  |
| 12 |          | TC     | P20LEMC1 |                                          |  |
| 13 |          |        |          |                                          |  |
| 14 | P20LEMD  | CAF    | 1500DEC  |                                          |  |
| 15 |          | TC     | TWIDDLE  | # WAITLIST FOR 15 SECONDS                |  |
| 16 |          | ADRES  | P20LEMD1 |                                          |  |
| 17 |          | TC     | ENDOFJOB |                                          |  |
| 18 |          |        |          |                                          |  |
| 19 | P20LEMD1 | CAE    | FLAGWRD1 | # IS TRACK FLAG SET                      |  |
| 20 |          | MASK   | TRACKBIT |                                          |  |
| 21 |          | CCS    | A        |                                          |  |
| 22 |          | TCF    | P20LEMD2 | # YES -- SCHEDULE DESIGNATE JOB          |  |
| 23 |          | TC     | FIXDELAY | # NO -- WAIT 15 SECONDS                  |  |
| 24 |          | DEC    | 1500     |                                          |  |
| 25 |          | TC     | P20LEMD1 |                                          |  |
| 26 |          |        |          |                                          |  |
| 27 | P20LEMD2 | CAF    | PRI026   | # SCHEDULE JOB TO DO R21                 |  |
| 28 |          | TC     | FINDVAC  |                                          |  |
| 29 |          | EBANK= | LOSCOUNT |                                          |  |
| 30 |          | 2CADR  | P20LEMC3 | # START AT PERM. MEMORY INTEGRATION      |  |
| 31 |          |        |          |                                          |  |
| 32 |          | TC     | TASKOVER |                                          |  |
| 33 |          |        |          |                                          |  |
| 34 | 250DEC   | DEC    | 250      |                                          |  |
| 35 | ALRM526  | OCT    | 00526    |                                          |  |
| 36 | OCT201   | OCT    | 00201    |                                          |  |
| 37 | ALRM514  | OCT    | 514      |                                          |  |
| 38 | MAXTRIES | DEC    | 60       |                                          |  |
| 39 | OCT00012 | OCT    | 00012    |                                          |  |
| 40 | P22ONE   | OCT    | 00001    |                                          |  |
| 41 | ONEB-2   | 2DEC   | 1.0 B-2  |                                          |  |
| 42 |          |        |          |                                          |  |
| 43 | V06N33*  | VN     | 0633     |                                          |  |
| 44 | UPPSV    | STQ    | CALL     | # UPDATES PERMANENT STATE VECTORS        |  |
| 45 |          |        | LS21X    | # TO PRESENT TIME                        |  |
| 46 |          |        | INTSTALL |                                          |  |
| 47 |          | CALL   |          |                                          |  |
| 48 |          |        |          |                                          |  |
| 49 |          |        |          |                                          |  |
| 50 |          |        |          |                                          |  |
| 51 |          |        |          |                                          |  |
| 52 |          |        |          |                                          |  |
| 53 |          |        |          |                                          |  |
| 54 |          |        |          |                                          |  |
| 55 |          |        |          |                                          |  |
| 56 |          |        |          |                                          |  |
| 57 |          |        |          |                                          |  |
| 58 |          |        |          |                                          |  |
| 59 |          |        |          |                                          |  |
| 60 |          |        |          |                                          |  |

|    |        |        |          |                                           |
|----|--------|--------|----------|-------------------------------------------|
| 1  |        |        | SETIFLG  |                                           |
| 2  |        |        | SET      |                                           |
| 3  |        | BOF    |          | # IF W-MATRIX INVALID, DON'T INTEGRATE IT |
| 4  |        |        | RENDWFLG |                                           |
| 5  |        |        | UPPSV1   |                                           |
| 6  |        |        | DIM0FLAG | # SET DIM0FLAG TO INTEGRATE W-MATRIX      |
| 7  |        | BON    | SET      |                                           |
| 8  |        |        | SURFFLAG | # IF ON LUNAR SURFACE W IS 6X6            |
| 9  |        |        | UPPSV5   |                                           |
| 10 |        |        | D6OR9FLG | # OTHERWISE 9X9                           |
| 11 | UPPSV5 | BOF    |          |                                           |
| 12 |        |        | VEHUPFLG |                                           |
| 13 |        |        | UPPSV3   |                                           |
| 14 | UPPSV1 | SET    |          |                                           |
| 15 |        |        | VINTFLAG |                                           |
| 16 |        | CALL   |          |                                           |
| 17 |        |        | INTEGRV  |                                           |
| 18 |        | CALL   |          | # GROUP 2 PHASE CHANGE                    |
| 19 |        |        | GRP2PC   | # TO PROTECT INTEGRATION                  |
| 20 |        | CALL   |          |                                           |
| 21 |        |        | INTSTALL |                                           |
| 22 |        | DLOAD  | CLEAR    | # GET TETCSM TO STORE IN TDEC FOR LM INT. |
| 23 |        |        | TETCSM   |                                           |
| 24 |        |        | VINTFLAG |                                           |
| 25 | UPPSV4 | CALL   |          | # INTEGRATE OTHER VEHICLE                 |
| 26 |        |        | SETIFLG  | # WITHOUT W-MATRIX                        |
| 27 |        | STCALL | TDEC1    |                                           |
| 28 |        |        | INTEGRV  |                                           |
| 29 |        | BOFF   | VLOAD    |                                           |
| 30 |        |        | SURFFLAG |                                           |
| 31 |        |        | P20LEMC4 |                                           |
| 32 |        |        | RCVLEM   |                                           |
| 33 |        | VSR2   |          |                                           |
| 34 |        | STOVL  | LMPOS    |                                           |
| 35 |        |        | VCVLEM   |                                           |
| 36 |        | VSR2   |          |                                           |
| 37 |        | STORE  | LMVEL    |                                           |
| 38 |        | GOTO   |          |                                           |
| 39 |        |        | LS21X    |                                           |
| 40 |        |        |          |                                           |
| 41 | UPPSV3 | CLEAR  | CALL     |                                           |
| 42 |        |        | VINTFLAG |                                           |
| 43 |        |        | INTEGRV  |                                           |
| 44 |        | CALL   |          |                                           |
| 45 |        |        | GRP2PC   |                                           |
| 46 |        | CALL   |          |                                           |
| 47 |        |        | INTSTALL |                                           |
| 48 |        | SET    | DLOAD    |                                           |
| 49 |        |        | VINTFLAG |                                           |
| 50 |        |        | TETLEM   | # GET TETLEM TO STORE IN TDEC FOR CSM INT |



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60

GOTO  
EBANK= UPPSV4  
COUNT\* LOSCOUNT  
          \$/P22



## # PROGRAM DESCRIPTION

# PREFERRED TRACKING ATTITUDE PROGRAM P25  
# MOD NO -- 3  
# BY P. VOLANTE

## # FUNCTIONAL DESCRIPTION

# THE PURPOSE OF THIS PROGRAM IS TO COMPUTE THE PREFERRED TRACKING  
# ATTITUDE OF THE LM TO CONTINUOUSLY POINT THE LM TRTACKING BEACON AT THE  
# CSM AND TO PERFORM THE MANEUVER TO THE PREFERRED TRACKING ATTITUDE AND  
# CONTINUOUSLY MAINTAIN THIS ATTITUDE WITHIN PRESCRIBED LIMITS.

## # CALLING SEQUENCE --

# ASTRONAUT REQUEST THROUGH DSKY V37E25E

## # SUBROUTINES CALLED --

|   |                            |          |
|---|----------------------------|----------|
| # | BANKCALL                   | FLAGUP   |
| # | R02BOTH (IMU STATUS CHECK) | ENDOFJOB |
| # | R61LEM (PREF TRK ATT ROUT) | WAITLIST |
| # | TASKOVER                   | FINDVAC  |

## # NORMAL EXIT MODES --

# P25 MAY BE TERMINATED IN TWO WAYS -- ASTRONAUT SELECTION OF IDLING  
# PROGRAM (P00) BY KEYING V37E00E OR BY KEYING IN V56E

## # ALARM OR ABORT EXIT MODES --

# NONE

## # OUTPUT

## # ERASABLE INITIALIZATION REQUIRED

## # FLAGS SET + RESET

# TRACKFLG, P25FLAG

## # DEBRIS

# NONE

|        |        |          |                                   |
|--------|--------|----------|-----------------------------------|
|        | EBANK= | LOSCOUNT |                                   |
|        | COUNT* | \$\$/P25 |                                   |
| PROG25 | TC     | 2PHSCHNG |                                   |
|        | OCT    | 4        | # MAKE GROUP 4 INACTIVE (VERB 37) |
|        | OCT    | 05022    |                                   |
|        | OCT    | 26000    | # PRIORITY 26                     |

|  |       |          |                    |
|--|-------|----------|--------------------|
|  | TC    | BANKCALL |                    |
|  | CADR  | R02BOTH  | # IMU STATUS CHECK |
|  | TC    | UPFLAG   |                    |
|  | ADRES | TRACKFLG | # SET TRACK FLAG   |

|         |       |          |               |
|---------|-------|----------|---------------|
|         | TC    | UPFLAG   |               |
|         | ADRES | P25FLAG  | # SET P25FLAG |
| P25LEM1 | TC    | PHASCHNG |               |





|    |          |        |          |                                             |  |    |
|----|----------|--------|----------|---------------------------------------------|--|----|
| 1  |          |        |          |                                             |  | 1  |
| 2  |          | BZF    | P25LMWT1 | # NO -- SKIP PHASE CHANGE AND WAIT 1 MINUTE |  | 2  |
| 3  |          | CAF    | SEVEN    | # CALL R65 -- FINE PREFERRED                |  | 3  |
| 4  |          | TS     | R65CNTR  |                                             |  | 4  |
| 5  |          | TC     | BANKCALL | # TRACKING ATTITUDE ROUTINE                 |  | 5  |
| 6  |          | CADR   | R65LEM   |                                             |  | 6  |
| 7  |          | TC     | P25LEM1  | # THEN GO CHECK FLAGS                       |  | 7  |
| 8  | P25LEMWT | TC     | PHASCHNG |                                             |  | 8  |
| 9  |          | OCT    | 00112    |                                             |  | 9  |
| 10 | P25LMWT1 | CAF    | 60SCNDS  |                                             |  | 10 |
| 11 |          | TC     | TWIDDLE  | # WAIT ONE MINUTE THEN CHECK AGAIN          |  | 11 |
| 12 |          | ADRES  | P25LEM2  |                                             |  | 12 |
| 13 |          | TC     | ENDOFJOB |                                             |  | 13 |
| 14 | P25LEM2  | CAF    | PRI014   |                                             |  | 14 |
| 15 |          | TC     | FINDVAC  |                                             |  | 15 |
| 16 |          | EBANK= | LOSCOUNT |                                             |  | 16 |
| 17 |          | 2CADR  | P25LEM1  |                                             |  | 17 |
| 18 |          |        |          |                                             |  | 18 |
| 19 |          | TC     | TASKOVER |                                             |  | 19 |
| 20 | 60SCNDS  | DEC    | 6000     |                                             |  | 20 |
| 21 |          |        |          |                                             |  | 21 |
| 22 |          |        |          |                                             |  | 22 |
| 23 |          |        |          |                                             |  | 23 |
| 24 |          |        |          |                                             |  | 24 |
| 25 |          |        |          |                                             |  | 25 |
| 26 |          |        |          |                                             |  | 26 |
| 27 |          |        |          |                                             |  | 27 |
| 28 |          |        |          |                                             |  | 28 |
| 29 |          |        |          |                                             |  | 29 |
| 30 |          |        |          |                                             |  | 30 |
| 31 |          |        |          |                                             |  | 31 |
| 32 |          |        |          |                                             |  | 32 |
| 33 |          |        |          |                                             |  | 33 |
| 34 |          |        |          |                                             |  | 34 |
| 35 |          |        |          |                                             |  | 35 |
| 36 |          |        |          |                                             |  | 36 |
| 37 |          |        |          |                                             |  | 37 |
| 38 |          |        |          |                                             |  | 38 |
| 39 |          |        |          |                                             |  | 39 |
| 40 |          |        |          |                                             |  | 40 |
| 41 |          |        |          |                                             |  | 41 |
| 42 |          |        |          |                                             |  | 42 |
| 43 |          |        |          |                                             |  | 43 |
| 44 |          |        |          |                                             |  | 44 |
| 45 |          |        |          |                                             |  | 45 |
| 46 |          |        |          |                                             |  | 46 |
| 47 |          |        |          |                                             |  | 47 |
| 48 |          |        |          |                                             |  | 48 |
| 49 |          |        |          |                                             |  | 49 |
| 50 |          |        |          |                                             |  | 50 |
| 51 |          |        |          |                                             |  | 51 |
| 52 |          |        |          |                                             |  | 52 |
| 53 |          |        |          |                                             |  | 53 |
| 54 |          |        |          |                                             |  | 54 |
| 55 |          |        |          |                                             |  | 55 |
| 56 |          |        |          |                                             |  | 56 |
| 57 |          |        |          |                                             |  | 57 |
| 58 |          |        |          |                                             |  | 58 |
| 59 |          |        |          |                                             |  | 59 |
| 60 |          |        |          |                                             |  | 60 |

# DATA READ ROUTINE 22 (LEM)  
# PROGRAM DESCRIPTION

#  
# MOD NO -- 2  
# BY P. VOLANTE

# FUNCTIONAL DESCRIPTION

# TO PROCESS AUTOMATIC RR MARK DATA TO UPDATE THE STATE VECTOR OF EITHER  
# LM OR CSM AS DEFINED IN THE RENDEZVOUS NAVIGATION PROGRAM (P20)

# CALLING SEQUENCE --

# TC BANKCALL  
# CADR R22LEM

# SUBROUTINES CALLED --

# LSR22.1 GOFLASH WAITLIST  
# LSR22.2 PRIOLARM BANKCALL  
# LSR22.3 R61LEM

# NORMAL EXIT MODES --

# R22 WILL CONTINUE TO RECYCLE, UPDATING STATE VECTORS WITH RADAR DATA  
# UNTIL P20 CEASES TO OPERATE (RENDEZVOUS FLAG SET TO ZERO) AT WHICH TIME  
# R22 WILL TERMINATE SELF.

# ALARM OR ABORT EXIT MODES --

# PRIORITY ALARM  
# PRIORITY ALARM 525 LOS NOT WITHIN 3 DEGREE LIMIT

# OUTPUT

# SEE OUTPUT FROM LSR22.3

# ERASABLE INITIALIZATION REQUIRED

# SEE LSR22.1, LSR22.2, LSR22.3

# FLAGS SET + RESET

# NOANGFLG

# DEBRIS

# SEE LSR22.1, LSR22.2, LSR22.3

|        |        |          |                           |
|--------|--------|----------|---------------------------|
|        | EBANK= | LRS22.1X |                           |
|        | COUNT* | \$\$/R22 |                           |
| R22LEM | TC     | PHASCHNG |                           |
|        | OCT    | 04022    |                           |
|        | CAF    | RNDVZBIT | # IS RENDEZVOUS FLAG SET? |
|        | MASK   | STATE    |                           |
|        | EXTEND |          |                           |
|        | BZF    | ENDOFJOB | # NO -- EXIT R22 AND P20  |
|        | CAF    | TRACKBIT | # IS TRACKFLAG SET?       |
|        | MASK   | STATE +1 |                           |

[illegible]

|    |         |        |          |                                              |
|----|---------|--------|----------|----------------------------------------------|
| 1  |         |        |          |                                              |
| 2  |         | EXTEND |          |                                              |
| 3  |         | BZF    | +2       | # YES CONTINUE                               |
| 4  |         | TC     | P20LEMB5 | # NO -- SET IT                               |
| 5  |         | CS     | RADMODES | # ARE RR CDUS BEING ZEROED                   |
| 6  |         | MASK   | RCDU0BIT |                                              |
| 7  |         | EXTEND |          |                                              |
| 8  |         | BZF    | R22LEM42 | # CDUS BEING ZEROED                          |
| 9  |         | TC     | PHASCHNG | # IF A RESTART OCCURS, AND EXTRA RADAR       |
| 10 |         | OCT    | 00152    | # READING IS TAKEN, SO BAD DATA ISN'T USED   |
| 11 |         | TC     | BANKCALL | # YES READ DATA + CALCULATE LOS              |
| 12 |         | CADR   | LRS22.1  | # DATA READ SUBROUTINE                       |
| 13 |         | INDEX  | MPAC     |                                              |
| 14 |         | TC     | +1       |                                              |
| 15 |         | TC     | R22LEM2  | # NORMAL RETURN (GOOD DATA)                  |
| 16 |         | TC     | P20LEMC  | # COULD NOT READ RADAR -- TRY TO REDESIGNATE |
| 17 |         | CAF    | ALRM525  | # RR LOS NOT WITHIN 3 DEGREES (ALARM)        |
| 18 |         | TC     | BANKCALL |                                              |
| 19 |         | CADR   | PRIOLARM |                                              |
| 20 |         | TC     | GOTOV56  | # TERMINATE EXITS P20 VIA V56 CODING         |
| 21 |         | TC     | R22LEM1  | # PROC (DISPLAY DELTA THETA)                 |
| 22 |         | TC     | -5       | # ENTER (ILLEGAL OPTION)                     |
| 23 |         | TC     | ENDOFJOB |                                              |
| 24 |         |        |          |                                              |
| 25 | R22LEM1 | TC     | PHASCHNG |                                              |
| 26 |         | OCT    | 04022    |                                              |
| 27 |         | CAF    | V06N05   | # DISPLAY DELTA THETA                        |
| 28 |         | TC     | BANKCALL |                                              |
| 29 |         | CADR   | PRIODSP  |                                              |
| 30 |         | TC     | GOTOV56  | # TERMINATE EXITS P20 VIA V56 CODING         |
| 31 |         | TC     | R22LEM2  | # PROC (OK CONTINUE)                         |
| 32 |         | TC     | P20LEMC  | # ENTER (RECYCLE)                            |
| 33 | R22LEM2 | TC     | PHASCHNG |                                              |
| 34 |         | OCT    | 04022    |                                              |
| 35 |         | TC     | LUNSFCHK | # CHECK IF ON LUNAR SURFACE (P22FLAG SET)    |
| 36 |         | TC     | R22LEM3  | # YES -- BYPASS FLAG CHECKS AND LRS22.2      |
| 37 |         | CA     | FLAGWRD1 | # IS TRACK FLAG SET                          |
| 38 |         | MASK   | TRACKBIT |                                              |
| 39 |         | EXTEND |          |                                              |
| 40 |         | BZF    | R22WAIT  | # NO -- WAIT                                 |
| 41 |         | TC     | BANKCALL | # YES                                        |
| 42 |         | CADR   | LRS22.2  | # CHECKS RR BORESIGHT WITHIN 30 DEG OF +Z    |
| 43 |         | INDEX  | MPAC     |                                              |
| 44 |         | TC     | +1       |                                              |
| 45 |         | TC     | R22LEM3  | # NORMAL RETURN (LOS WITHIN 30 OF Z-AXIS)    |
| 46 |         | TC     | BANKCALL |                                              |
| 47 |         | CADR   | R61LEM   |                                              |
| 48 |         | TC     | R22WAIT  | # NOT WITHIN 30 DEG OF Z-AXIS                |
| 49 | R22LEM3 | CS     | FLAGWRD1 | # SHOULD WE BYPASS STATE VECTOR UPDATE       |
| 50 |         | MASK   | NOUPFBIT | # (IS NO UPDATE FLAG SET?)                   |
| 51 |         |        |          |                                              |
| 52 |         |        |          |                                              |
| 53 |         |        |          |                                              |
| 54 |         |        |          |                                              |
| 55 |         |        |          |                                              |
| 56 |         |        |          |                                              |
| 57 |         |        |          |                                              |
| 58 |         |        |          |                                              |
| 59 |         |        |          |                                              |
| 60 |         |        |          |                                              |

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
|          | EXTEND |          |                                           |
|          | BZF    | R22LEM42 | # BRANCH -- YES                           |
|          | CA     | FLAGWRD1 | # IS UPDATE FLAG SET                      |
|          | MASK   | UPDATBIT |                                           |
|          | EXTEND |          |                                           |
|          | BZF    | R22LEM42 | # UPDATE FLAG NOT SET                     |
|          | CAF    | PRI026   | # INSURE HIGH PRIO IN RESTART             |
|          | TS     | PHSPRDT2 |                                           |
|          | TC     | INTPRET  |                                           |
|          | GOTO   |          |                                           |
|          |        | LSR22.3  |                                           |
| R22LEM93 | EXIT   |          | # NORMAL EXIT FROM LSR22.3                |
|          | TC     | PHASCHNG | # PHASE CHANGE TO PROTECT AGAINST         |
|          | OCT    | 04022    | # CONFLICT WITH GRP2PC ERASEABLE          |
|          | TCF    | R22LEM44 |                                           |
| R22LEM96 | EXIT   |          |                                           |
|          | CAF    | ZERO     | # SET N49FLAG = ZERO TO INDICATE          |
|          | TS     | N49FLAG  | # V06 N49 DISPLAY HASN'T BEEN ANSWERED    |
|          | TC     | PHASCHNG |                                           |
|          | OCT    | 04022    | # TO PROTECT DISPLAY                      |
|          | CAF    | PRI027   | # PROTECT DISPLAY                         |
|          | TC     | NOVAC    |                                           |
|          | EBANK= | N49FLAG  |                                           |
|          | 2CADR  | N49DSP   |                                           |
|          | TC     | INTPRET  |                                           |
|          | SLOAD  |          |                                           |
|          |        | N49FLAG  |                                           |
|          | BZE    | BMN      | # LOOP TO CHECK IF FLAG                   |
|          |        | -3       | # SETTING CHANGED -- BRANCH -- NO         |
|          |        | R22LEM7  | # PROCEED                                 |
|          | EXIT   |          | # DISPLAY ANSERED BY RECYCLE              |
|          | TC     | LUNSFCHK | # ARE WE ON LUNAR SURFACE                 |
|          | TC     | R22WAIT  | # YES -- 15 SECOND DELAY                  |
|          | CA     | ZERO     | # NO -- SET R65COUNTER = 0, DO FINE       |
|          | TC     | R22LEM45 | # TRACKING TAKE ANOTHER RADAR READING     |
| R22LEM7  | CALL   |          | # PROCEED                                 |
|          |        | GRP2PC   | # PHASE CHANGE AND                        |
|          | GOTO   |          | # GO TO INCOPORATE DATA.                  |
|          |        | ASTOK    |                                           |
| R22LEM44 | INCR   | MARKCTR  | # INCREMENT COUNT OF MARKS INCORPORATED.  |
|          | TC     | LUNSFCHK | # ARE WE ON LUNAR SURFACE                 |
|          | TC     | R22LEM46 | # YES -- WAIT 2 SECONDS                   |
|          | CA     | FIVE     | # NOT ON LUNAR SURFACE                    |
|          | TC     | R22LEM45 | # R65COUNTER = 5                          |
| R22LEM42 | TC     | LUNSFCHK | # CHECK IF ON LUNAR SURFACE (P22FLAG SET) |
|          | TC     | R22LEM46 | # YES -- WAIT 2 SECONDS                   |
|          | CA     | TWO      | # NO -- SET R65COUNTER = 2                |
| R22LEM45 | TS     | R65CNTR  |                                           |

|            |         |                                                          |                                             |
|------------|---------|----------------------------------------------------------|---------------------------------------------|
|            | TC      | BANKCALL                                                 |                                             |
|            | CADR    | R65LEM                                                   | # FINE PREFERRED TRACKING ATTITUDE          |
| R22WAIT    | TC      | R22LEM                                                   |                                             |
|            | CAF     | 1500DEC                                                  |                                             |
|            | TC      | P20LEMWT +1                                              |                                             |
| R22LEM46   | CAF     | 2SECS                                                    |                                             |
|            | TC      | BANKCALL                                                 | # WAIT 2 SECONDS AND TAKE ANOTHER MARK      |
|            | CADR    | DELAYJOB                                                 |                                             |
|            | TC      | R22LEM                                                   |                                             |
| N49DSP     | CAF     | V06N49NB                                                 |                                             |
|            | TC      | BANKCALL                                                 | # EXCESSIVE STATE VECTOR UPDATE -- FLASH    |
|            | CADR    | PRIODSP                                                  | # VERB 06 NOUN 49 R1=DELTA R, R2=DELTA V    |
|            | TC      | GOTOV56                                                  | # TERMINATE -- EXIT R22 AND P20             |
|            | CS      | ONE                                                      | # PROCEED -- N49FLAG = -1                   |
|            | TS      | N49FLAG                                                  | # RECYCLE -- N49FLAG = + VALUE              |
| R22RSTRT   | TC      | ENDOFJOB                                                 |                                             |
|            | TC      | PHASCHNG                                                 | # IF A RESTART OCCURS WHILE READING RADAR   |
|            | OCT     | 00152                                                    | # COME HERE TO TAKE A RANGE-RATE READING    |
|            | TC      | BANKCALL                                                 | # WHICH ISN'T USED TO PREVENT TAKING A BAD  |
|            | CADR    | RRRDOT                                                   | # READING AND TRYING TO INCORPORATE THE     |
|            | TC      | BANKCALL                                                 | # BAD DATA                                  |
|            | CADR    | RADSTALL                                                 | # WAIT FOR READ COMPLETE                    |
|            | TC      | P20LEMC                                                  | # COULD NOT READ RADAR -- TRY TO REDISGNATE |
|            | TC      | R22LEM                                                   | # READ SUCCESSFUL -- CONTINUE AT R22        |
| ALRM525    | OCT     | 00525                                                    |                                             |
| V06N05     | VN      | 00605                                                    |                                             |
| V06N49NB   | VN      | 00649                                                    |                                             |
| 1500DEC    | DEC     | 1500                                                     |                                             |
| # LUNSFCHK | --      | CLOSED SUBROUTINE TO CHECK IF ON LUNAR SURFACE (P22FLAG) |                                             |
| #          | RETURNS | TO CALLER +1 IF P22FLAG SET                              |                                             |
| #          |         | TO CALLER +2 IF P22FLAG NOT SET                          |                                             |
| LUNSFCHK   | COUNT*  | \$\$/P22                                                 |                                             |
|            | CS      | FLAGWRD8                                                 | # CHECK IF ON LUNAR SURFACE                 |
|            | MASK    | SURFFBIT                                                 | # IS SURFFLAG SET?                          |
|            | CCS     | A                                                        | # BRANCH -- P22FLAG SET                     |
|            | INCR    | Q                                                        | # NOT SET                                   |
|            | TC      | Q                                                        | # RETURN                                    |



# RR DESIGNATE ROUTINE (R21LEM)  
# PROGRAM DESCRIPTION

#  
# MOD NO -- 2  
# BY P. VOLANTE

# FUNCTIONAL DESCRIPTION

# TO POINT THE RENDEZVOUS RADAR AT THE CSM UNTIL AUTOMATIC ACQUISITION  
# OF THE CSM IS ACCOMPLISHED BY THE RADAR. ROUTINE IS CALLED BY P20.

# CALLING SEQUENCE --

# TC BANKCALL  
# CADR R21LEM

# SUBROUTINES CALLED --

|   |          |          |          |          |
|---|----------|----------|----------|----------|
| # | FINDVAC  | FLAGUP   | ENDOFJOB | PRIOLARM |
| # | NOVAC    | INTPRET  | LPS20.1  | PHASCHNG |
| # | WAITLIST | JOBSLEEP | JOBWAKE  | FLAGDOWN |
| # | TASKOVER | BANKCALL | RADSTALL | RRDESSM  |

# NORMAL EXIT MODES

# WHEN LOCK-ON IS ACHIEVED, BRANCH WILL BE TO P20 WHERE R22 (DATA READ  
# WILL BE SELECTED OR A NEED FOR A MANEUVER (BRANCH TO P20LEMA)

# ALARM OR ABORT EXIT MODES --

# PRIORITY ALARM 503 WHEN LOCK-ON HASN'T BEEN ACHIEVED AFTER 30SECS --  
# THIS REQUIRES ASTRONAUT INTERFACE: SELECTION OF SEARCH OPTION OF  
# ACQUISITION

# OUTPUT

# SEE LPS20.1, RRDESSM

# ERASABLE INITIALIZATION REQUIRED

# RRTARGET, RADMODES ARE USED BY LPS20.1 AND RRDESSM

# FLAGS SET + RESET

# LOSCMFLG LOKONSW

# DEBRIS

# SEE LPS20.1, RRSESSM

|        |        |          |                               |
|--------|--------|----------|-------------------------------|
|        | EBANK= | LOSCOUNT |                               |
|        | COUNT* | \$\$/R21 |                               |
| R21LEM | CS     | BIT14    | # REMOVE RR SELF TRACK ENABLE |

|  |        |          |
|--|--------|----------|
|  | EXTEND |          |
|  | WAND   | CHAN12   |
|  | TC     | LUNSFCHK |

|  |     |         |                                            |
|--|-----|---------|--------------------------------------------|
|  | TC  | R21LEM5 |                                            |
|  | CAF | ZERO    | # COMMAND ANTENNA TO MODE CENTER           |
|  | TS  | TANG    | # IF NOT ON SURFACE -- MODE 1 -- (T=0,S=0) |

|    |         |      |          |    |
|----|---------|------|----------|----|
|    |         | TS   | TANG +1  |    |
| 1  | R21LEM5 | TC   | R21LEM6  | 1  |
| 2  |         | CAF  | BIT12    | 2  |
| 3  |         | MASK | RADMODES | 3  |
| 4  |         | CCS  | A        | 4  |
| 5  |         | TC   | R21LEM10 | 5  |
| 6  |         | CAF  | BIT15    | 6  |
| 7  |         | TS   | TANG     | 7  |
| 8  |         | CS   | HALF     | 8  |
| 9  |         | TS   | TANG +1  | 9  |
| 10 |         |      |          | 10 |
| 11 |         |      |          | 11 |
| 12 |         |      |          | 12 |
| 13 |         |      |          | 13 |
| 14 |         |      |          | 14 |
| 15 |         |      |          | 15 |
| 16 |         |      |          | 16 |
| 17 |         |      |          | 17 |
| 18 |         |      |          | 18 |
| 19 |         |      |          | 19 |
| 20 |         |      |          | 20 |
| 21 |         |      |          | 21 |
| 22 |         |      |          | 22 |
| 23 |         |      |          | 23 |
| 24 |         |      |          | 24 |
| 25 |         |      |          | 25 |
| 26 |         |      |          | 26 |
| 27 |         |      |          | 27 |
| 28 |         |      |          | 28 |
| 29 |         |      |          | 29 |
| 30 |         |      |          | 30 |
| 31 |         |      |          | 31 |
| 32 |         |      |          | 32 |
| 33 |         |      |          | 33 |
| 34 |         |      |          | 34 |
| 35 |         |      |          | 35 |
| 36 |         |      |          | 36 |
| 37 |         |      |          | 37 |
| 38 |         |      |          | 38 |
| 39 |         |      |          | 39 |
| 40 |         |      |          | 40 |
| 41 |         |      |          | 41 |
| 42 |         |      |          | 42 |
| 43 |         |      |          | 43 |
| 44 |         |      |          | 44 |
| 45 |         |      |          | 45 |
| 46 |         |      |          | 46 |
| 47 |         |      |          | 47 |
| 48 |         |      |          | 48 |
| 49 |         |      |          | 49 |
| 50 |         |      |          | 50 |
| 51 |         |      |          | 51 |
| 52 |         |      |          | 52 |
| 53 |         |      |          | 53 |
| 54 |         |      |          | 54 |
| 55 |         |      |          | 55 |
| 56 |         |      |          | 56 |
| 57 |         |      |          | 57 |
| 58 |         |      |          | 58 |
| 59 |         |      |          | 59 |
| 60 |         |      |          | 60 |

|          |        |          |                                              |
|----------|--------|----------|----------------------------------------------|
| R21LEM6  | TC     | DOWNFLAG |                                              |
|          | ADRES  | LOKONSW  |                                              |
|          | TC     | BANKCALL |                                              |
|          | CADR   | RRDESNB  |                                              |
|          | TC     | +1       |                                              |
|          | TC     | BANKCALL |                                              |
|          | CADR   | RADSTALL |                                              |
|          | TC     | R21-503  | # BAD RETURN FROM DESIGNATE -- ISSUE ALARM   |
| R21LEM10 | TC     | UPFLAG   |                                              |
|          | ADRES  | LOSCMFLG | # EVERY FOURTH PASS THRU DODES               |
|          | CAF    | MAXTRIES | # ALLOW 60 PASSES (APPROX 45 SECONDS)        |
| R21LEM2  | TS     | DESCOUNT | # TO DESIGNATE AND LOCK ON                   |
|          | CAF    | THREE    |                                              |
|          | TS     | LOSCOUNT |                                              |
| R21LEM1  | TC     | INTPRET  |                                              |
|          | RTB    | DAD      |                                              |
|          |        | LOADTIME |                                              |
|          | STCALL | HALFSEC  | # EXTRAPOLATE TO PRESENT TIME + .5 SEC.      |
|          |        | TDEC1    | # LOS DETERMINATION ROUTINE                  |
|          |        | LPS20.1  |                                              |
| R21LEM3  | EXIT   |          |                                              |
|          | TC     | UPFLAG   | # SET LOKONSW TO RADAR -- ON DESIRED         |
|          | ADRES  | LOKONSW  |                                              |
|          | TC     | DOWNFLAG |                                              |
|          | ADRES  | NORRMON  |                                              |
|          | TC     | INTPRET  |                                              |
|          | CALL   |          | # INPUT (RRTARGET UPDATED BY LPS20.1)        |
|          |        | RRDESSM  | # DESIGNATE ROUTINE                          |
|          | EXIT   |          |                                              |
|          | TC     | R21LEM4  | # LOS NOT IN MODE 2 COVERAGE                 |
|          |        |          | # ON LUNAR SURFACE                           |
|          | TC     | P20LEMA  | # VEHICLE MANEUVER REQUIRED.                 |
|          | TC     | BANKCALL | # NO VEHICLE MANEUVER REQUIRED               |
|          | CADR   | RADSTALL | # WAIT FOR DESIGNATE COMPLETE -- LOCKON OR   |
|          | TC     | +2       | # BAD END -- LOCKON NOT ACHIEVED IN 60 TRIES |
| R21-503  | TC     | R21END   | # EXIT ROUTINE RETURN TO P20 (LOCK-ON)       |
|          | CAF    | ALRM503  | # ISSUE ALARM 503                            |
|          | TC     | BANKCALL |                                              |
|          | CADR   | PRIOLARM |                                              |
|          | TC     | GOTOV56  | # TERMINATE EXITS P20 VIA V56 CODING         |
|          | TC     | R21SRCH  | # PROC                                       |
|          | TC     | P20LEMC3 |                                              |
|          | TC     | ENDOFJOB |                                              |
| R21END   | TC     | DOWNFLAG |                                              |
|          | ADRES  | LOSCMFLG | # RESET LOSCMFLG                             |
|          | TC     | R21DISP  | # PUT UP VERIFY MAIN LOBE LOCKON DISPLAY     |
| R21SRCH  | TC     | PHASCHNG |                                              |
|          | OCT    | 04022    |                                              |
|          | TC     | R24LEM   | # SEARCH ROUTINE                             |
| ALRM503  | OCT    | 00503    |                                              |

|    |          |        |             |                                    |
|----|----------|--------|-------------|------------------------------------|
| 1  | ALRM527  | OCT    | 527         |                                    |
| 2  |          |        |             |                                    |
| 3  |          |        |             |                                    |
| 4  | R21LEM4  | CAF    | MAXTRIES    | # SET UP COUNTER FOR               |
| 5  |          | TS     | REPOSCNT    | # 60 PASSES (APPROX 600 SECS.)     |
| 6  |          | TC     | UPFLAG      |                                    |
| 7  |          | ADRES  | FSPASFLG    | # SET FIRST PASS FLAG              |
| 8  |          | TC     | DOWNFLAG    | # RESET LOS BEING                  |
| 9  |          | ADRES  | LOSCMFLG    | # COMPUTED FLAG                    |
| 10 |          | TC     | INTPRET     |                                    |
| 11 | R21LEM12 | RTB    |             |                                    |
| 12 |          |        | LOADTIME    |                                    |
| 13 |          | DAD    |             |                                    |
| 14 |          |        | TENSEC      | # TIME T = T + 10 SECS.            |
| 15 |          | STORE  | REPOSTM     | # SAVE FOR LONGCALL AND UPPSV      |
| 16 |          | STCALL | TDEC1       |                                    |
| 17 |          |        | LPS20.1     | # COMPUTE LOS AT TIME T            |
| 18 |          | CALL   |             |                                    |
| 19 |          |        | RRDESSM     |                                    |
| 20 |          | EXIT   |             |                                    |
| 21 |          | TC     | R21LEM13    | # LOS NOT IN MODE 2 COVERAGE       |
| 22 |          | TC     | ENDOFJOB    | # VEHICLE MANEUVER REQUIRED        |
| 23 |          | TC     | KILLTASK    |                                    |
| 24 |          | CADR   | BEGDES      |                                    |
| 25 |          | TC     | INTPRET     |                                    |
| 26 |          | BOF    |             |                                    |
| 27 |          |        | FSPASFLG    | # FIRST PASS THRU REPOSITION       |
| 28 |          |        | R21LEMB     | # NO -- GO TO CONTINUOUS DESIGNATE |
| 29 |          | CLRGO  |             |                                    |
| 30 |          |        | FSPASFLG    | # YES -- RESET FIRST PASS FLAG     |
| 31 |          |        | R21LEM7 +1  |                                    |
| 32 | R21LEM13 | CCS    | REPOSCNT    | # HAVE WE TRIED 60 TIMES?          |
| 33 |          | TC     | R21LEM7     | # NO -- ADD 10 SECS. RECOMPUTE LOS |
| 34 |          | TC     | R21LEM11    | # YES -- PUT OUT ALARM 530         |
| 35 | R21LEM7  | TS     | REPOSCNT    |                                    |
| 36 |          | TC     | INTPRET     |                                    |
| 37 |          | DLOAD  | GOTO        |                                    |
| 38 |          |        | REPOSTM     |                                    |
| 39 |          |        | R21LEM12 +2 |                                    |
| 40 | R21LEMB  | DLOAD  |             |                                    |
| 41 |          |        | REPOSTM     |                                    |
| 42 |          | STCALL | TDEC1       |                                    |
| 43 |          |        | UPPSV       |                                    |
| 44 |          | EXIT   |             |                                    |
| 45 |          | TC     | UPFLAG      | # SET RADMODES BIT 15 FOR          |
| 46 |          | ADRES  | CDESFLAG    | # CONTINUOUS DESIGNATION           |
| 47 |          | TC     | DOWNFLAG    |                                    |
| 48 |          | ADRES  | LOKONSW     |                                    |
| 49 |          | TC     | UPFLAG      |                                    |
| 50 |          | ADRES  | NORRMON     |                                    |
| 51 |          |        |             |                                    |
| 52 |          |        |             |                                    |
| 53 |          |        |             |                                    |
| 54 |          |        |             |                                    |
| 55 |          |        |             |                                    |
| 56 |          |        |             |                                    |
| 57 |          |        |             |                                    |
| 58 |          |        |             |                                    |
| 59 |          |        |             |                                    |
| 60 |          |        |             |                                    |

|    |          |        |           |                                            |  |
|----|----------|--------|-----------|--------------------------------------------|--|
| 1  |          |        |           |                                            |  |
| 2  |          | TC     | BANKCALL  |                                            |  |
| 3  |          | CADR   | RRDESNB   |                                            |  |
| 4  |          | TC     | +1        |                                            |  |
| 5  |          | TC     | INTPRET   |                                            |  |
| 6  |          | RTB    | BDSU      |                                            |  |
| 7  |          |        | LOADTIME  | # COMPUTE DELTA TIME                       |  |
| 8  |          |        | REPOSTM   | # FOR LONGCALL                             |  |
| 9  |          | STORE  | DELTATM   |                                            |  |
| 10 |          | EXIT   |           |                                            |  |
| 11 |          | EXTEND |           |                                            |  |
| 12 |          | DCA    | DELTATM   |                                            |  |
| 13 |          | TC     | LONGCALL  |                                            |  |
| 14 |          | EBANK= | LOSCOUNT  |                                            |  |
| 15 |          | 2CADR  | R21LEM9   |                                            |  |
| 16 |          |        |           |                                            |  |
| 17 |          | TC     | ENDOFJOB  |                                            |  |
| 18 | R21LEM9  | TC     | KILLTASK  |                                            |  |
| 19 |          | CADR   | STDESIG   |                                            |  |
| 20 |          | TC     | CLRADMOD  |                                            |  |
| 21 |          | CAF    | PRI026    |                                            |  |
| 22 |          | TC     | FINDVAC   |                                            |  |
| 23 |          | EBANK= | LOSCOUNT  |                                            |  |
| 24 |          | 2CADR  | R21LEM10  |                                            |  |
| 25 |          |        |           |                                            |  |
| 26 |          | TC     | TASKOVER  |                                            |  |
| 27 | R21LEM11 | CAF    | ALRM530   | # ALARM 530 -- LOS NOT IN COVERAGE         |  |
| 28 |          | TC     | BANKCALL  | # AFTER TRYING TO DESIGNATE FOR            |  |
| 29 |          | CADR   | PRIOLARM  | # 600 SECS.                                |  |
| 30 |          | TC     | GOTOV56   |                                            |  |
| 31 |          | TC     | GOTOV56   |                                            |  |
| 32 |          | TC     | GOTOV56   |                                            |  |
| 33 |          | TC     | ENDOFJOB  |                                            |  |
| 34 | ALRM530  | OCT    | 00530     |                                            |  |
| 35 | TENSEC   | 2DEC   | 1000 B-28 |                                            |  |
| 36 |          |        |           |                                            |  |
| 37 | HALFSEC  | 2DEC   | 50        |                                            |  |
| 38 |          |        |           |                                            |  |
| 39 | R21DISP  | TC     | PHASCHNG  |                                            |  |
| 40 |          | OCT    | 04022     |                                            |  |
| 41 |          | CAF    | V06N72PV  | # FLASH V 50 N 72 -- PLEASE PERFORM RR     |  |
| 42 |          | TC     | BANKCALL  | # MAIN LOBE LOCKON VERIFICATION            |  |
| 43 |          | CADR   | GOPERF2R  |                                            |  |
| 44 |          | TC     | GOTOV56   | # TERMINATE EXITS VIA V 56                 |  |
| 45 |          | TC     | P20LEMWT  | # PROCEED CONTINUES TO R22                 |  |
| 46 |          | TC     | -5        | # ENTER ILLEGAL                            |  |
| 47 |          | CAF    | BIT7      |                                            |  |
| 48 |          | TC     | LINUS     | # SET BITS TO MAKE THIS A PRIORITY DISPLAY |  |
| 49 |          | TC     | ENDOFJOB  |                                            |  |
| 50 |          |        |           |                                            |  |
| 51 |          |        |           |                                            |  |
| 52 |          |        |           |                                            |  |
| 53 |          |        |           |                                            |  |
| 54 |          |        |           |                                            |  |
| 55 |          |        |           |                                            |  |
| 56 |          |        |           |                                            |  |
| 57 |          |        |           |                                            |  |
| 58 |          |        |           |                                            |  |
| 59 |          |        |           |                                            |  |
| 60 |          |        |           |                                            |  |



|    |          |  |  |    |       |  |    |
|----|----------|--|--|----|-------|--|----|
| 1  | V06N72PV |  |  | VN | 00672 |  | 1  |
| 2  |          |  |  |    |       |  | 2  |
| 3  |          |  |  |    |       |  | 3  |
| 4  |          |  |  |    |       |  | 4  |
| 5  |          |  |  |    |       |  | 5  |
| 6  |          |  |  |    |       |  | 6  |
| 7  |          |  |  |    |       |  | 7  |
| 8  |          |  |  |    |       |  | 8  |
| 9  |          |  |  |    |       |  | 9  |
| 10 |          |  |  |    |       |  | 10 |
| 11 |          |  |  |    |       |  | 11 |
| 12 |          |  |  |    |       |  | 12 |
| 13 |          |  |  |    |       |  | 13 |
| 14 |          |  |  |    |       |  | 14 |
| 15 |          |  |  |    |       |  | 15 |
| 16 |          |  |  |    |       |  | 16 |
| 17 |          |  |  |    |       |  | 17 |
| 18 |          |  |  |    |       |  | 18 |
| 19 |          |  |  |    |       |  | 19 |
| 20 |          |  |  |    |       |  | 20 |
| 21 |          |  |  |    |       |  | 21 |
| 22 |          |  |  |    |       |  | 22 |
| 23 |          |  |  |    |       |  | 23 |
| 24 |          |  |  |    |       |  | 24 |
| 25 |          |  |  |    |       |  | 25 |
| 26 |          |  |  |    |       |  | 26 |
| 27 |          |  |  |    |       |  | 27 |
| 28 |          |  |  |    |       |  | 28 |
| 29 |          |  |  |    |       |  | 29 |
| 30 |          |  |  |    |       |  | 30 |
| 31 |          |  |  |    |       |  | 31 |
| 32 |          |  |  |    |       |  | 32 |
| 33 |          |  |  |    |       |  | 33 |
| 34 |          |  |  |    |       |  | 34 |
| 35 |          |  |  |    |       |  | 35 |
| 36 |          |  |  |    |       |  | 36 |
| 37 |          |  |  |    |       |  | 37 |
| 38 |          |  |  |    |       |  | 38 |
| 39 |          |  |  |    |       |  | 39 |
| 40 |          |  |  |    |       |  | 40 |
| 41 |          |  |  |    |       |  | 41 |
| 42 |          |  |  |    |       |  | 42 |
| 43 |          |  |  |    |       |  | 43 |
| 44 |          |  |  |    |       |  | 44 |
| 45 |          |  |  |    |       |  | 45 |
| 46 |          |  |  |    |       |  | 46 |
| 47 |          |  |  |    |       |  | 47 |
| 48 |          |  |  |    |       |  | 48 |
| 49 |          |  |  |    |       |  | 49 |
| 50 |          |  |  |    |       |  | 50 |
| 51 |          |  |  |    |       |  | 51 |
| 52 |          |  |  |    |       |  | 52 |
| 53 |          |  |  |    |       |  | 53 |
| 54 |          |  |  |    |       |  | 54 |
| 55 |          |  |  |    |       |  | 55 |
| 56 |          |  |  |    |       |  | 56 |
| 57 |          |  |  |    |       |  | 57 |
| 58 |          |  |  |    |       |  | 58 |
| 59 |          |  |  |    |       |  | 59 |
| 60 |          |  |  |    |       |  | 60 |

# MANUAL ACQUISITION ROUTINE R23LEM  
# PROGRAM DESCRIPTION

#  
# MOD NO -- 2  
# BY P. VOLANTE

#  
# FUNCTIONAL DESCRIPTION

# TO ACQUIRE THE CSM BY MANUAL OPERATION OF THE RENDEZVOUS RADAR  
#  
# CALLING SEQUENCE --

#  
# TC R23LEM

# SUBROUTINES CALLED

# BANKCALL R61LEM  
# SETMINDB GOPERF1

# NORMAL EXIT MODES --

#  
# IN RESPONSE TO THE GOPERF1, SELECTION OF ENTER WILL RECYCLE R23  
# SELECTION OF PROC WILL CONTINUE R23  
# SELECTION OF TERM WILL TERMINATE R23 + P20

# ALARM OR ABORT EXIT MODES --

#  
# SEE NORMAL EXIT MODES ABOVE

# OUTPUT

#  
# N.A.

# ERASABLE INITIALIZATION REQUIRED --

#  
# ACMODFLG MUST BE SET TO 1 (MANUAL MODE)

#  
# EBANK= GENRET  
# COUNT\* \$\$/R23  
# R23LEM TC UPFLAG # SET NO ANGLE MONITOR FLAG  
# ADRES NORRMON

#  
# INHINT  
# TC IBNKCALL # SELECT MINIMUM DEADBAND  
# CADR SETMINDB

#  
# R23LEM1 RELINT  
# CAF BIT14 # ENABLE TRACKER  
# EXTEND

#  
# WOR CHAN12  
# CAF OCT205  
# TC BANKCALL

#  
# CADR GOPERF1  
# TC R23LEM2 # TERMINATE  
# TC R23LEM11 # PROCEDE  
# R23LEM11 TC R23LEM3 # ENTER -- DO ANOTHER MANEUVER

#  
# R23LEM11 INHINT  
# TC RRLIMCHK # YES -- CHECK IF ANTENNA IS WITHIN LIMITS

#  
# ADRES CDUT  
# TC OUTOFLIM # NOT WITHIN LIMITS  
# TC IBNKCALL # RESTORE DEADBAND TO



1412THE

1

|    |          |          |                                      |
|----|----------|----------|--------------------------------------|
|    | CADR     | RESTORDB | # ASTRONAUT SELECTED VALUE           |
| 1  | RELINT   |          |                                      |
| 2  | TC       | DOWNFLAG | # CLEAR NO ANGLE MONITOR FLAG        |
| 3  | ADRES    | NORRMON  |                                      |
| 4  | TC       | P20LEMB1 | # RADAR IS LOCKED ON CONTINUE IN P20 |
| 5  | OUTOFLIM | RELINT   |                                      |
| 6  |          |          |                                      |
| 7  |          |          |                                      |
| 8  |          |          |                                      |
| 9  |          |          |                                      |
| 10 |          |          |                                      |
| 11 |          |          |                                      |
| 12 |          |          |                                      |
| 13 |          |          |                                      |
| 14 |          |          |                                      |
| 15 |          |          |                                      |
| 16 |          |          |                                      |
| 17 |          |          |                                      |
| 18 |          |          |                                      |
| 19 |          |          |                                      |
| 20 |          |          |                                      |
| 21 |          |          |                                      |
| 22 |          |          |                                      |
| 23 |          |          |                                      |
| 24 |          |          |                                      |
| 25 |          |          |                                      |
| 26 |          |          |                                      |
| 27 |          |          |                                      |
| 28 |          |          |                                      |
| 29 |          |          |                                      |
| 30 |          |          |                                      |
| 31 |          |          |                                      |
| 32 |          |          |                                      |
| 33 |          |          |                                      |
| 34 |          |          |                                      |
| 35 |          |          |                                      |
| 36 |          |          |                                      |
| 37 |          |          |                                      |
| 38 |          |          |                                      |
| 39 |          |          |                                      |
| 40 |          |          |                                      |
| 41 |          |          |                                      |
| 42 |          |          |                                      |
| 43 |          |          |                                      |
| 44 |          |          |                                      |
| 45 |          |          |                                      |
| 46 |          |          |                                      |
| 47 |          |          |                                      |
| 48 |          |          |                                      |
| 49 |          |          |                                      |
| 50 |          |          |                                      |
| 51 |          |          |                                      |
| 52 |          |          |                                      |
| 53 |          |          |                                      |
| 54 |          |          |                                      |
| 55 |          |          |                                      |
| 56 |          |          |                                      |
| 57 |          |          |                                      |
| 58 |          |          |                                      |
| 59 |          |          |                                      |
| 60 |          |          |                                      |



[illegible]

# SEARCH ROUTINE R24LEM  
# PROGRAM DESCRIPTION

#  
# MOD NO -- 2  
# BY P. VOLANTE

#  
# FUNCTIONAL DESCRIPTION

#  
# TO ACQUIRE THE CSM BY A SEARCH PATTERN WHEN THE RENDEZVOUS RADAR HAS  
# FAILED TO ACQUIRE TEH CSM IN THE AUTOMATIC TRACKING MODE AND TO ALLOW  
# THE ASTRONAUT TO CONFIRM THAT REACQUISITION HAS NOT BEEN IN SIDELobe.

#  
# CALLING SEQUENCE

#  
# CAF PRIONN  
# TC FINDVAC  
# EBANK= DATAGOOD  
# 2CADR R24LEM

#  
# SUBROUTINES CALLED

#  
# FLAGUP FLAGDOWN BANKCALL  
# R61LEM GOFLASHR FINDVAC  
# ENDOFJOB NOVAC LSR24.1

#  
# NORMAL EXIT MODES --

#  
# ASTRONAUT RESPONSE TO DISPLAY OF OMEGA AND DATAGOOD. HE CAN EITHER  
# REJECT BY TERMINATING (SEARCH OPTION AND RESELECTING P20) OR ACCEPT BY  
# PROCEEDING (EXIT ROUTINE AND RETURN TO AUTO MODE IN P20)

#  
# ALARM OR ABORT EXIT MODES --

#  
# SEE NORMAL EXIT MODES ABOVE

#  
# OUTPUT --

#  
# SEE OUTPUT FROM LSR24.1 + R61LEM

#  
# ERASABLE INITIALIZATION REQUIRED

#  
# SET INPUT FOR LSR24.1

#  
# FLAGS SET + RESET

#  
# SRCHOPT, ACMODFLG

|         |        |           |                                         |
|---------|--------|-----------|-----------------------------------------|
|         | EBANK= | DATAGOOD  |                                         |
|         | COUNT* | \$\$/R24  |                                         |
| R24LEM  | TC     | UPFLAG    |                                         |
|         | ADRES  | SRCHOPTN  | # SET SRCHOPT FLAG                      |
|         | TC     | DOWNFLAG  | # RESET LOS BEING COMPUTED FLAG TO MAKE |
|         | ADRES  | LOSCMFLG  | # SURE DODES DOESN'T GO TO R21          |
| R24LEM1 | CAF    | ZERO      |                                         |
|         | TS     | DATAGOOD  | # ZERO OUT DATA INDICATOR               |
|         | TS     | OMEGAD    | # ZERO OMEGA DISPLAY REGS               |
|         | TS     | OMEGAD +1 | # ZERO OMEGA DISPLAY REGS               |
| R24LEM2 | TC     | PHASCHNG  |                                         |
|         | OCT    | 04022     |                                         |

|    |  |      |          |                                       |    |
|----|--|------|----------|---------------------------------------|----|
|    |  | CAF  | V16N80   |                                       |    |
| 1  |  | TC   | BANKCALL |                                       | 1  |
| 2  |  | CADR | PRIODSPR |                                       | 2  |
| 3  |  | TC   | GOTOV56  |                                       | 3  |
| 4  |  | TC   | R24END   | # PROCEED EXIT R24 TO P20LEM1         | 4  |
| 5  |  |      |          |                                       | 5  |
| 6  |  | TC   | R24LEM3  | # RECYCLE -- CALL R61 TO MANEUVER S/C | 6  |
| 7  |  |      |          |                                       | 7  |
| 8  |  |      |          |                                       | 8  |
| 9  |  |      |          |                                       | 9  |
| 10 |  |      |          |                                       | 10 |
| 11 |  |      |          |                                       | 11 |
| 12 |  |      |          |                                       | 12 |
| 13 |  |      |          |                                       | 13 |
| 14 |  |      |          |                                       | 14 |
| 15 |  |      |          |                                       | 15 |
| 16 |  |      |          |                                       | 16 |
| 17 |  |      |          |                                       | 17 |
| 18 |  |      |          |                                       | 18 |
| 19 |  |      |          |                                       | 19 |
| 20 |  |      |          |                                       | 20 |
| 21 |  |      |          |                                       | 21 |
| 22 |  |      |          |                                       | 22 |
| 23 |  |      |          |                                       | 23 |
| 24 |  |      |          |                                       | 24 |
| 25 |  |      |          |                                       | 25 |
| 26 |  |      |          |                                       | 26 |
| 27 |  |      |          |                                       | 27 |
| 28 |  |      |          |                                       | 28 |
| 29 |  |      |          |                                       | 29 |
| 30 |  |      |          |                                       | 30 |
| 31 |  |      |          |                                       | 31 |
| 32 |  |      |          |                                       | 32 |
| 33 |  |      |          |                                       | 33 |
| 34 |  |      |          |                                       | 34 |
| 35 |  |      |          |                                       | 35 |
| 36 |  |      |          |                                       | 36 |
| 37 |  |      |          |                                       | 37 |
| 38 |  |      |          |                                       | 38 |
| 39 |  |      |          |                                       | 39 |
| 40 |  |      |          |                                       | 40 |
| 41 |  |      |          |                                       | 41 |
| 42 |  |      |          |                                       | 42 |
| 43 |  |      |          |                                       | 43 |
| 44 |  |      |          |                                       | 44 |
| 45 |  |      |          |                                       | 45 |
| 46 |  |      |          |                                       | 46 |
| 47 |  |      |          |                                       | 47 |
| 48 |  |      |          |                                       | 48 |
| 49 |  |      |          |                                       | 49 |
| 50 |  |      |          |                                       | 50 |
| 51 |  |      |          |                                       | 51 |
| 52 |  |      |          |                                       | 52 |
| 53 |  |      |          |                                       | 53 |
| 54 |  |      |          |                                       | 54 |
| 55 |  |      |          |                                       | 55 |
| 56 |  |      |          |                                       | 56 |
| 57 |  |      |          |                                       | 57 |
| 58 |  |      |          |                                       | 58 |
| 59 |  |      |          |                                       | 59 |
| 60 |  |      |          |                                       | 60 |

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
|          | TC     | BANKCALL |                                            |
|          | CADR   | LRS24.1  |                                            |
| R24END   | TC     | KILLTASK |                                            |
|          | CADR   | CALLDGCH |                                            |
|          | TC     | CLRADM0D | # CLEAR BITS 10 & 15 OF RADMODES.          |
|          | TCF    | P20LEM1  | # AND GO TO 400 MI. RANGE CHECK IN P20     |
|          | BLOCK  | 3        |                                            |
|          | SETLOC | FFTAG6   |                                            |
|          | BANK   |          |                                            |
|          | COUNT* | \$\$/R24 |                                            |
| CLRADM0D | CS     | BIT10+15 |                                            |
|          | INHINT |          |                                            |
|          | MASK   | RADMODES |                                            |
|          | TS     | RADMODES |                                            |
|          | CS     | BIT2     | # DISABLE RR ERROR COUNTERS                |
|          | EXTEND |          |                                            |
|          | WAND   | CHAN12   | # USER WILL RELINT                         |
|          | TC     | Q        |                                            |
| BIT10+15 | OCT    | 41000    |                                            |
|          | BANK   | 24       |                                            |
|          | SETLOC | P20S     |                                            |
|          | BANK   |          |                                            |
|          | COUNT* | \$\$/R24 |                                            |
| R24LEM3  | TC     | PHASCHNG |                                            |
|          | OCT    | 04022    |                                            |
|          | TC     | KILLTASK |                                            |
|          | CADR   | CALLDGCH | # KILL WAITLIST FOR NEXT POINT IN PATTERN  |
|          | TC     | CLRADM0D | # CLEAR BITS 10 + 15 OF RADMODES TO KILL   |
|          | RELINT |          | # HALF SECOND DESIGNATE LOOP               |
|          | CAF    | .5SEC    |                                            |
|          | TC     | BANKCALL | # WAIT FOR DESIGNATE LOOP TO DIE           |
|          | CADR   | DELAYJOB |                                            |
|          | TC     | LUNSFCHK | # CHECK IF ON LUNAR SURFACE                |
|          | TC     | R24LEM4  | # YES -- DON'T DO ATTITUDE MANEUVER        |
|          | TC     | BANKCALL | # CALL R61 TO DO PREFERRED TRACKING        |
|          | CADR   | R61LEM   | # ATTITUDE MANEUVER                        |
| R24LEM4  | CAF    | ZERO     | # ZERO OUT RADCADR (WHICH WAS SET BY       |
|          | TS     | RADCADR  | # ENDRADAR WHEN DESIGNATE STOPPED) SO THAT |
|          |        |          | # RRDESSM WILL RETURN TO CALLER            |
|          | TC     | R24LEM2  | # AND GO BACK TO PUT UP V16 N80 DISPLAY    |
| V16N80   | VN     | 01680    |                                            |

# PREFERRED TRACKING ATTITUDE ROUTINE R61LEM  
# PROGRAM DESCRIPTION

# MOD NO: 3 DATE: 4-11-67  
# MOD BY: P. VOLANTE, SDC

# FUNCTIONAL DESCRIPTION --

# TO COMPUTE THE PREFERRED TRACKING ATTITUDE OF THE LM TO ENABLE RR  
# TRACKING OF THE CSM AND TO PERFORM THE MANEUVER TO THE PREFERRED  
# ATTITUDE.

# CALLING SEQUENCE --

# TC BANKCALL  
# CADR R61LEM

# SUBROUTINES CALLED

# LPS20.1 VECPOINT  
# KALCMAN3

# NORMAL EXIT MODES --

# NORMAL RETURN IS TO CALLER + 1

# ALARM OR ABORT EXIT MODES --

# TERMINATE P20 + R61 BY BRANCHING TO P20END IF BOTH TRACKFLAG +  
# RENDEZVOUS FLAG ARE NOT SET.

# OUTPUT --

# SEE OUTPUT FOR LPS20.1 + ATTITUDE MANEUVER ROUTINE (R60)

# ERASABLE INITIALIZATION REQUIRED

# GENRET USED TO SAVE Q FOR RETURN

# FLAGS SET + RESET

# 3AXISFLG

# DEBRIS

# SEE SUBROUTINES

|        |        |          |                  |
|--------|--------|----------|------------------|
|        | SETLOC | R61      |                  |
|        | BANK   |          |                  |
|        | EBANK= | LOSCOUNT |                  |
|        | COUNT* | \$\$/R61 |                  |
| R61LEM | TC     | MAKECADR |                  |
|        | TS     | GENRET   |                  |
|        | TC     | UPFLAG   | # SET R61 FLAG   |
|        | ADRES  | R61FLAG  |                  |
|        | TC     | R61C+L01 |                  |
| R65LEM | TC     | MAKECADR |                  |
|        | TS     | GENRET   |                  |
|        | TC     | DOWNFLAG | # RESET R61 FLAG |



|    |          |        |          |                                            |
|----|----------|--------|----------|--------------------------------------------|
| 1  |          |        | HIUNITZ  |                                            |
| 2  |          |        | SCAXIS   | # TRACK AXIS UNIT VECTOR                   |
| 3  |          | STORE  |          |                                            |
| 4  | R61LEM1  | RTB    | DAD      |                                            |
| 5  |          |        | LOADTIME | # EXTRAPOLATE FORWARD TO CENTER            |
| 6  |          |        | 3SECONDS | # SIX SECOND PERIOD.                       |
| 7  |          | STCALL | TDEC1    |                                            |
| 8  |          |        | LPS20.1  | # LOS DETERMINATION + VEH ATTITUDE         |
| 9  |          | VLOAD  |          |                                            |
| 10 |          |        | RRTARGET |                                            |
| 11 |          | STORE  | POINTVSM |                                            |
| 12 |          | RTB    | CALL     | # GET DESIRED CDU'S FOR VECPNT1            |
| 13 |          |        | READCDUD |                                            |
| 14 |          |        | VECPNT1  | # COMPUTES FINAL ANGLES FROM PRESENT CDUDS |
| 15 |          | STORE  | CPHI     | # STORE FINAL ANGLES -- CPHI, CTHETA, CPSI |
| 16 |          | EXIT   |          |                                            |
| 17 |          | TC     | PHASCHNG |                                            |
| 18 |          | OCT    | 04022    |                                            |
| 19 |          | CAF    | TRACKBIT | # IS TRACK FLAG SET                        |
| 20 |          | MASK   | FLAGWRD1 |                                            |
| 21 |          | EXTEND |          |                                            |
| 22 |          | BZF    | R65WAIT  |                                            |
| 23 |          | TC     | BANKCALL |                                            |
| 24 |          | CADR   | G+N,AUTO | # CHECK FOR AUTO MODE                      |
| 25 |          | CCS    | A        |                                            |
| 26 |          | TC     | R61C+L04 | # NOT IN AUTO                              |
| 27 |          | TC     | INTPRET  |                                            |
| 28 |          | VLOAD  | CALL     |                                            |
| 29 |          |        | RRTARGET |                                            |
| 30 |          |        | CDU*SMNB |                                            |
| 31 |          | DLOAD  | DSU      | # GET PHI -- ARCCOS OF Z-COMPONENT OF LOS  |
| 32 |          |        | MPAC +5  |                                            |
| 33 |          |        | COS15DEG |                                            |
| 34 | R61LEM2  | BMN    | EXIT     | # BRANCH -- PHI > 15 DEGREES               |
| 35 |          |        | R61C+L05 | # PHI GRE 10DEG                            |
| 36 |          | EBANK= | CDUXD    |                                            |
| 37 |          | CAF    | EBANK6   |                                            |
| 38 |          | TS     | EBANK    |                                            |
| 39 |          | INHINT |          |                                            |
| 40 |          | EXTEND |          |                                            |
| 41 |          | DCA    | CPHI     |                                            |
| 42 |          | DXCH   | CDUXD    |                                            |
| 43 |          | CA     | CPSI     |                                            |
| 44 |          | TS     | CDUZD    |                                            |
| 45 |          | RELINT |          |                                            |
| 46 |          | EBANK= | LOSCOUNT |                                            |
| 47 |          | CAF    | EBANK7   |                                            |
| 48 |          | TS     | EBANK    |                                            |
| 49 |          | TC     | R61C+L06 |                                            |
| 50 | R61C+L05 | EXIT   |          |                                            |
| 51 |          | INHINT |          |                                            |

|    |          |        |          |                                   |
|----|----------|--------|----------|-----------------------------------|
| 1  |          |        |          |                                   |
| 2  |          | TC     | IBNKCALL |                                   |
| 3  |          | FCADR  | ZATTEROR |                                   |
| 4  |          | TC     | IBNKCALL |                                   |
| 5  |          | FCADR  | SETMINDB | # REDUCE ATTITUDE ERROR           |
| 6  |          | TC     | DOWNFLAG |                                   |
| 7  |          | ADRES  | 3AXISFLG |                                   |
| 8  |          | TC     | UPFLAG   |                                   |
| 9  |          | ADRES  | PDSPFLAG | # SET PRIORITY DISPLAY FLAG       |
| 10 |          | TC     | BANKCALL |                                   |
| 11 |          | CADR   | R60LEM   |                                   |
| 12 |          | INHINT |          |                                   |
| 13 |          | TC     | IBNKCALL |                                   |
| 14 |          | FCADR  | RESTORDB |                                   |
| 15 |          | TC     | PHASCHNG |                                   |
| 16 |          | OCT    | 04022    |                                   |
| 17 |          | TC     | DOWNFLAG |                                   |
| 18 |          | ADRES  | PDSPFLAG | # RESET PRIORITY DISPLAY FLAG     |
| 19 | R61C+L06 | CA     | FLAGWRD1 |                                   |
| 20 |          | MASK   | R61FLBIT |                                   |
| 21 |          | CCS    | A        |                                   |
| 22 |          | TC     | R61C+L4  |                                   |
| 23 |          | CCS    | R65CNTR  |                                   |
| 24 |          | TC     | +2       |                                   |
| 25 |          | TC     | R61C+L4  | # R65CNTR = 0 - EXIT ROUTINE      |
| 26 |          | TS     | R65CNTR  |                                   |
| 27 |          | CAF    | 06SEC    |                                   |
| 28 |          | TC     | TWIDDLE  |                                   |
| 29 |          | ADRES  | R61C+L2  |                                   |
| 30 |          | TC     | ENDOFJOB |                                   |
| 31 | R61C+L2  | CAF    | PRI026   |                                   |
| 32 |          | TC     | FINDVAC  |                                   |
| 33 |          | EBANK= | LOSCOUNT |                                   |
| 34 |          | 2CADR  | R61C+L01 |                                   |
| 35 |          |        |          |                                   |
| 36 |          | TC     | TASKOVER |                                   |
| 37 | R61C+L04 | TC     | BANKCALL | # TO CONVERT ANGLES TO FDAI       |
| 38 |          | CADR   | BALLANGS |                                   |
| 39 |          | TC     | R61C+L06 |                                   |
| 40 | R61C+L4  | CAE    | GENRET   |                                   |
| 41 |          | TCF    | BANKJUMP | # EXIT R61                        |
| 42 | R61C+L1  | CAF    | BIT7+9PV | # IS RENDEZVOUS OR P25FLAG SET    |
| 43 |          | MASK   | STATE    |                                   |
| 44 |          | EXTEND |          |                                   |
| 45 |          | BZF    | ENDOFJOB | # NO -- EXIT ROUTINE AND PROGRAM. |
| 46 |          | TC     | R61C+L06 | # YES EXIT ROUTINE                |
| 47 | R65WAIT  | TC     | POSTJUMP |                                   |
| 48 |          | CADR   | P20LEMWT |                                   |
| 49 |          |        |          |                                   |
| 50 | BIT7+9PV | OCT    | 00500    |                                   |
| 51 |          |        |          |                                   |
| 52 |          |        |          |                                   |
| 53 |          |        |          |                                   |
| 54 |          |        |          |                                   |
| 55 |          |        |          |                                   |
| 56 |          |        |          |                                   |
| 57 |          |        |          |                                   |
| 58 |          |        |          |                                   |
| 59 |          |        |          |                                   |
| 60 |          |        |          |                                   |



COS15DEG      2DEC      0.96593   B-1

|          |        |     |                                     |
|----------|--------|-----|-------------------------------------|
| 06SEC    | DEC    | 600 |                                     |
| PHI      | EQUALS | 20D |                                     |
| READCDUD | INHINT |     | # READS DESIRED CDU'S AND STORES IN |

```
READS DESIRED CDU'S AND STORES IN
MPAC TP EXITS WITH MODE SET TO TP
```

|     |          |
|-----|----------|
| CAF | EBANK6   |
| XCH | EBANK    |
| TS  | RUPTREG1 |

|        |       |
|--------|-------|
| EBANK= | CDUXD |
| CA     | CDUXD |
| TS     | MPAC  |

EXTEND

|      |         |
|------|---------|
| DCA  | CDUYD   |
| DXCH | MPAC +1 |

|    |          |
|----|----------|
| CA | RUPTREG1 |
| TS | EBANK    |

RELINT

| TCF | TMODE |
|-----|-------|
|-----|-------|

```
BLOCK 02
SETLOC RADARFF
```

BANK

|        |            |
|--------|------------|
| EBANK= | LOSCOUNT   |
| COUNT* | \$\$/RRSUB |

# THE FOLLOWING SUBROUTINE RETURNS TO CALLER +2 IF THE ABSOLUTE VALUE OF VALUE OF C(A) IS GREATER THAN THE  
# NEGATIVE OF THE NUMBER AT CALLER +1. OTHERWISE IT RETURNS TO CALLER +3. MAY BE CALLED IN RUPT OR UNDER EXEC.

MAGSUB

EXTEND

BZMF +2

TCF +2

COM

INDEX Q

AD 0

EXTEND

BZMF Q+2 # ABS(A) <= CONST GO TO L+3

TCF Q+1 # ABS(A) > CONST GO TO L+2

# PROGRAM NAME: RRLIMCHK

#

# FUNCTIONAL DESCRIPTION:

#

# RRLIMCHK CHECKS RR DESIRED GIMBAL ANGLES TO SEE IF THEY ARE WITHIN  
# THE LIMITS OF THE CURRENT MODE. INITIALLY THE DESIRED TRUNNION AND  
# SHAFT ANGLES ARE STORED IN ITEMP1 AND ITEMP2. THE CURRENT RR  
# ANTENNAE MODE (RADMODES BIT 12) IS CHECKED WHICH IS = 0 FOR  
# MODE 1 AND =1 FOR MODE 2.

#

# MODE 1 -- THE TRUNNION ANGLE IS CHECKED AT MAGSUB TO SEE IF IT IS  
# BETWEEN -55 AND +55 DEGREES. IF NOT, RETURN TO L +2. IF WITHIN LIMITS,  
# THE SHAFT ANGLE IS CHECKED TO SEE IF IT IS BETWEEN -70 AND +59 DEGREES.  
# IF NOT, RETURN TO L +2. IF IN LIMITS, RETURN TO L +3.

#

# MODE 2 -- THE SHAFT ANGLE IS CHECKED AT MAGSUB TO SEE IF IT IS  
# BETWEEN -139 AND -25 DEGREES. IF NOT, RETURN TO L +2. IF WITHIN  
# LIMITS, THE TRUNNION ANGLE IS CHECKED TO SEE IF IT IS BETWEEN +125  
# AND -125 (+235) DEGREES. IF NOT, RETURN TO L +2. IF IN LIMITS, RETURN  
# TO L +3.

#

# CALLING SEQUENCE:

#

# L TC RRLIMCHK (WITH INTERRUPT INHIBITED)  
# L +1 ADRES T,S (DESIRED TRUNNION ANGLE ADDRESS)

#

# ERASABLE INITIALIZATION REQUIRED:

#

# RADMODES, MODEA, MODEB (OR DESIRED TRUNNION AND SHAFT  
# ANGLES ELSEWHERE IN CONSECUTIVE LOCATIONS -- UNSWITCHED ERASABLE OR  
# CURRENT EBANK).

#

# SUBROUTINES CALLED: MAGSUB

#

# JOBS OR TASKS INITIATED: NONE

#

# ALARMS: NONE

#

# EXIT: L + 2 (EITHER OR BOTH ANGLES NOT WITHIN LIMITS OF CURRENT MODE)  
# L + 3 (BOTH ANGLES WITHIN LIMITS OF CURRENT MODE)

RRLIMCHK

EXTEND

INDEX Q

INDEX 0

DCA 0

INCR Q

DXCH ITEMP1

LXCH Q # L(CALLER +2) TO L.

CAF ANTENBIT # SEE WHICH MODE RR IS IN.

MASK RADMODES

CCS A

TCF MODE2CHK

CA ITEMP1 # MODE 1 IS DEFINED AS

[illegible]

# PROGRAM NAME: SETTRKF

# FUNCTIONAL DESCRIPTION:

# SETTRKF UPDATES THE TRACKER FAIL LAMP ON THE DSKY.

# INITIALLY THE LAMP TEST FLAG (IMODES33 BIT 1) IS CHECKED.

# IF A LAMP TEST IS IN PROGRESS, THE PROGRAM EXITS TO L +1.

# IF NO LAMP TEST THE FOLLOWING IS CHECKED SEQUENTIALLY:

# 1) RR CDU'S BEING ZEROED, RR CDU OK, AND RR NOT IN  
# AUTO MODE (RADMODES BITS 13, 7, 2).# 2) LR VEL DATA FAIL AND NO LR POS DATA (RADMODES BITS  
# 8,5)

# 3) NO RR DATA (RADMODES BIT 4)

# THE ABSENCE OF ALL THREE SIMULTANEOUSLY IN (1), THE PRESENCE OF BOTH

# IN (2), AND THE PRESENCE OF (3) RESULTS IN EITHER THE TRACKER FAIL

# LAMP (DSPTAB +11D BIT 8) BEING TURNED OFF OR IS LEFT OFF. THEREFORE, THE

# TRACKER FAIL LAMP IS TURN ON IF:

# A) RR CDU FAILED WITH RR IN AUTO MODE AND RR CDU'S NOT BEING ZEROED

# B) N SAMPLES OF LR DATA COULD NOT BE TAKEN IN 2N TRIES WITH  
# EITHER THE ALT OR VEL INFORMATION# C) N SAMPLES OF RR DATA COULD NOT BE OBTAINED FROM 2N TRIES  
# WITH EITHER THE AL

# CALLING SEQUENCE:

# L TC SETTRKF

# ERASABLE INITIALIZATION REQUIRED: IMODES33, RADMODES, DSPTAB +11D

# SUBROUTINES CALLED: NONE

# JOBS OR TASKS INITIATED: NONE

# ALARMS: TRACKER FAIL LAMP

# EXIT: L +1 (ALWAYS)

SETTRKF CAF BIT1 # NO ACTION IF DURING LAMP TEST

MASK IMODES33

CCS A

TC Q

RRTRKF CA BIT8

TS L

CAF 13,7,2 # SEE IF CDU FAILED.

MASK RADMODES

EXTEND TRKFLON # CONDITION 3 ABOVE.

BZF

RRCHECK CAF RRDATA BT # SEE IF RR DATA FAILED.

MASK RADMODES



|    |                    |        |                                             |    |
|----|--------------------|--------|---------------------------------------------|----|
| 1  |                    |        |                                             | 1  |
| 2  | TRKFLON            | CCS    | A                                           | 2  |
| 3  |                    | CA     | L                                           | 3  |
| 4  |                    | AD     | DSPTAB +11D                                 | 4  |
| 5  |                    | MASK   | # HALF ADD DESIRED AND PRESENT STATES.      | 5  |
| 6  |                    | EXTEND | L                                           | 6  |
| 7  | FLIP               | BZF    | TCQ                                         | 7  |
| 8  |                    |        | # NO CHANGE.                                | 8  |
| 9  |                    | CA     | DSPTAB +11D                                 | 9  |
| 10 |                    | EXTEND | # CAN'T USE LXCH DSPTAB +11D (RESTART PROB) | 10 |
| 11 |                    | RXOR   | LCHAN                                       | 11 |
| 12 |                    | MASK   | POSMAX                                      | 12 |
| 13 |                    | AD     | BIT15                                       | 13 |
| 14 |                    | TS     | DSPTAB +11D                                 | 14 |
| 15 |                    | TC     | Q                                           | 15 |
| 16 | 13,7,2<br>ENDRMODF | OCT    | 10102                                       | 16 |
| 17 |                    | EQUALS |                                             | 17 |
| 18 |                    |        |                                             | 18 |
| 19 |                    |        |                                             | 19 |
| 20 |                    |        |                                             | 20 |
| 21 |                    |        |                                             | 21 |
| 22 |                    |        |                                             | 22 |
| 23 |                    |        |                                             | 23 |
| 24 |                    |        |                                             | 24 |
| 25 |                    |        |                                             | 25 |
| 26 |                    |        |                                             | 26 |
| 27 |                    |        |                                             | 27 |
| 28 |                    |        |                                             | 28 |
| 29 |                    |        |                                             | 29 |
| 30 |                    |        |                                             | 30 |
| 31 |                    |        |                                             | 31 |
| 32 |                    |        |                                             | 32 |
| 33 |                    |        |                                             | 33 |
| 34 |                    |        |                                             | 34 |
| 35 |                    |        |                                             | 35 |
| 36 |                    |        |                                             | 36 |
| 37 |                    |        |                                             | 37 |
| 38 |                    |        |                                             | 38 |
| 39 |                    |        |                                             | 39 |
| 40 |                    |        |                                             | 40 |
| 41 |                    |        |                                             | 41 |
| 42 |                    |        |                                             | 42 |
| 43 |                    |        |                                             | 43 |
| 44 |                    |        |                                             | 44 |
| 45 |                    |        |                                             | 45 |
| 46 |                    |        |                                             | 46 |
| 47 |                    |        |                                             | 47 |
| 48 |                    |        |                                             | 48 |
| 49 |                    |        |                                             | 49 |
| 50 |                    |        |                                             | 50 |
| 51 |                    |        |                                             | 51 |
| 52 |                    |        |                                             | 52 |
| 53 |                    |        |                                             | 53 |
| 54 |                    |        |                                             | 54 |
| 55 |                    |        |                                             | 55 |
| 56 |                    |        |                                             | 56 |
| 57 |                    |        |                                             | 57 |
| 58 |                    |        |                                             | 58 |
| 59 |                    |        |                                             | 59 |
| 60 |                    |        |                                             | 60 |

# PROGRAM NAME: RRTURNON

# FUNCTIONAL DESCRIPTION:

# RRTURNON IS THE TURN-ON SEQUENCE WHICH, ALONG WITH  
 # RRZEROSB, ZEROES THE CDU'S AND DETERMINES THE RR MODE.  
 # INITIALLY, CONTROL IS TRANSFERRED TO RRZEROSB FOR THE  
 # ACTUAL TURN-ON SEQUENCE. UPON RETURN THE PROGRAM  
 # WAITS 1 SECOND BEFORE REMOVING THE TURN-ON FLAG  
 # (RADMODES BIT1) SO THE REPOSITION ROUTINE WON'T  
 # INITIATE PROGRAM ALARM 00501. A CHECK IS THEN MADE  
 # TO SEE IF A PROGRAM IS USING THE RR (STATE BIT 7). IF  
 # SO, THE PROGRAM EXITS TO ENDRADAR SO THAT THE RR CDU  
 # FAIL FLAG (RADMODES BIT 7) CAN BE CHECKED BEFORE  
 # RETURNING TO THE WAITING PROGRAM. IF NOT, THE PROGRAM EXITS  
 # TO TASKOVER.

# CALLING SEQUENCE: WAITLIST TASK FROM RRAUTCHK IF THE RR POWER-ON AUTO  
 # BIT (CHAN 33 BIT 2) CHANGES TO 0 AND NO PROGRAM WAS USING  
 # THE RR (STATE BIT 7).

# ERASABLE INITIALIZATION REQUIRED:

# RADMODES, STATE

# SUBROUTINES CALLED: RRZEROSB, FIXDELAY, TASKOVER, ENDRADAR

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: NONE (SEE RRZEROSB)

# EXIT: TASKOVER, ENDRADAR (WAITING PROGRAM)

BANK 24  
 SETLOC P20S1  
 BANK

EBANK= LOSCOUNT  
 COUNT\* \$\$/RSUB

RRTURNON

TC RRZEROSB  
 TC FIXDELAY # WAIT 1 SEC BEFORE REMOVING TURN ON FLAG  
 DEC 100 # SO A MONITOR REPOSITION WON'T ALARM.

CS TURNONBT  
 MASK RADMODES  
 TS RADMODES  
 TCF TASKOVER

# PROGRAM NAME: RRZEROSB

# FUNCTIONAL DESCRIPTION:

# RRZEROSB IS A CLOSED SUBROUTINE TO ZERO THE RR CDU'S,  
# DETERMINE THE RR MODE, AND TURN ON THE TRACKER FAIL  
# LAMP IF REQUIRED. INITIALLY THE RR CDU ZERO BIT (CHAN 12  
# BIT 1) IS SET. FOLLOWING A 20 MILLISECOND WAIT, THE LGC  
# RR CDU COUNTERS (OPTY, OPTX) ARE SET = 0 AFTER  
# WHICH THE RR CDU ZERO DISCRETE (CHAN 12 BIT 1) IS  
# REMOVED. A 4 SECOND WAIT IS SET TO ALL THE RR CDU'S  
# TO REPEAT THE ACTUAL TRUNNION AND SHAFT ANGLES. THE  
# RR CDU ZERO FLAG (RADMODES BIT 13) IS REMOVED. THE  
# CONTENTS OF OPTY IS THEN CHECKED TO SEE IF THE TRUNNION  
# ANGLE IS LESS THAN 90 DEGREES. IF NOT, BIT 12 OF  
# RADMODES IS SET = 1 TO INDICATE RR ANTENNA MODE 2.  
# IF LESS THAN 90 DEGREES, BIT 12 OF RADMODES IS SET = 0 TO  
# INDICATE RR ANTENNA MODE 1. SETTRKF IS THEN CALLED TO  
# SEE IF THE TRACKER FAIL LAMP SHOULD BE TURNED ON.

# CALLING SEQUENCE: L TC RRZEROSB (FROM RRTURNON AND RRZERO)

# ERASABLE INITIALIZATION REQUIRED:

# RADMODES (BIT 13 SET), DSPTAB +11D

# SUBROUTINES CALLED: FIXDELAY, MAGSUB, SETTRKF

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: TRAKCER FAIL

# EXIT: L +1 (ALWAYS)

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| RRZEROSB | EXTEND |          |                                           |
|          | QXCH   | RRRET    |                                           |
|          | CAF    | BIT1     | # BIT 13 OF RADMODES MUST BE SET BEFORE   |
|          | EXTEND |          | # COMING HERE.                            |
|          | WOR    | CHAN12   | # TURN ON ZERO RR CDU                     |
|          | TC     | FIXDELAY |                                           |
|          | DEC    | 2        |                                           |
|          | CAF    | ZERO     |                                           |
|          | TS     | CDUT     |                                           |
|          | TS     | CDUS     |                                           |
|          | CS     | ONE      | # REMOVE ZEROING BIT.                     |
|          | EXTEND |          |                                           |
|          | WAND   | CHAN12   |                                           |
|          | TC     | FIXDELAY |                                           |
|          | DEC    | 1000     | # RESET FAIL INHIBIT IN 10 SECS. -- D.281 |
|          | CS     | RCDU0BIT | # REMOVE ZEROING IN PROCESS BIT           |



-BIT12

# PROGRAM NAME: DORREPOS

# FUNCTIONAL DESCRIPTION:

# DORREPOS IS A SEQUENCE OF TASKS TO DRIVE THE RENDEZVOUS RADAR TO A SAFE POSITION. INITIALLY SETRRECR IS CALLED WHERE THE RR ERROR COUNTERS (CHAN 12 BIT 2) ARE ENABLED AND LASTYCND AND LASTXCMD SET = 0 TO INDICATE THE DIFFERENCE BETWEEN THE DESIRED STATE AND PRESENT STATE OF THE COMMANDS. THE RR TURN-ON FLAG (RADMODES BIT 1) IS CHECKED AND IF NOT PRESENT, PROGRAM ALARM 00501 IS REQUESTED BEFORE CONTINUING. IN EITHER CASE, FOLLOWING A 20 MILLISECOND WAIT THE PROGRAM CHECKS THE CURRENT RR ANTENNA MODE (RADMODES BIT 12). RRONLY IS THEN CALLED TO DRIVE THE TRUNNION ANGLE TO 0 DEGREES IF IN MODE 1 AND TO 180 DEGREES IF IN MODE 2. UPON RETURN, THE CURRENT RR ANTENNA MODE (RADMODES BIT 12) IS AGAIN CHECKED. RRONLY IS THEN CALLED TO DRIVE THE SHAFT ANGLE TO 0 DEGREES IF IN MODE 1 AND TO -90 DEGREES IF IN MODE 2. IF DURING RRONLY OR RRONLY A REMODE HAS BEEN REQUESTED (RADMODES BIT 14), AND ALWAYS FOLLOWING COMPLETION OF RRONLY, CONTROL IS TRANSFERRED TO REPOSRT. HERE THE REPOSITION FLAG (RADMODES BIT 11) IS REMOVED. A CHECK IS THEN MADE ON THE DESIGNATE FLAG (RADMODES BIT 10). IF PRESENT, CONTROL IS TRANSFERRED TO BEGDES. IF NOT PRESENT INDICATING NO FURTHER ANTENNA CONTROL REQUIRED, THE RR ERROR COUNTER BIT (CHAN 12 BIT 2) IS REMOVED AND THE ROUTINE EXITS TO TASKOVER.

# CALLING SEQUENCE:

# WAITLIST CALL FROM RRGIMON IF TRUNNION AND SHAFT CDU ANGLES NOT WITHIN LIMITS OF CURRENT MODE.

# ERASABLE INITIALIZATION REQUIRED:

# RADMODES

# SUBROUTINES CALLED

# RRONLY, RRONLY, BEGDES (EXIT)

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: NONE

# EXIT: TASKOVER, BEGDES

|          |    |          |                                 |
|----------|----|----------|---------------------------------|
| DORREPOS | TC | SETRRECR | # SET UP RR CDU ERROR COUNTERS. |
|----------|----|----------|---------------------------------|

# ALARM 501 DELETED IN DANCE 279 PER PCR 97.

|     |          |
|-----|----------|
| TC  | FIXDELAY |
| DEC | 2        |

|     |          |                                           |
|-----|----------|-------------------------------------------|
| CAF | ANTENBIT | # MANEUVER TRUNNION ANGLE TO NOMINAL POS. |
|-----|----------|-------------------------------------------|

|    |           |        |          |                                           |    |
|----|-----------|--------|----------|-------------------------------------------|----|
| 1  | # 120 123 |        |          | PAGE 330                                  | 1  |
| 2  |           | MASK   | RADMODES |                                           | 2  |
| 3  |           | CCS    | A        |                                           | 3  |
| 4  |           | CAF    | BIT15    | # 0 FOR MODE 1 AND 180 FOR MODE 2.        | 4  |
| 5  |           | TC     | RRONLY   |                                           | 5  |
| 6  |           |        |          |                                           | 6  |
| 7  |           | CAF    | ANTENBIT | # NOT PUT SHAFT IN RIGHT POSITION         | 7  |
| 8  |           | MASK   | RADMODES |                                           | 8  |
| 9  |           | CCS    | A        |                                           | 9  |
| 10 |           | CS     | HALF     | # -90 FOR MODE 2.                         | 10 |
| 11 |           | TC     | RRONLY   |                                           | 11 |
| 12 |           |        |          |                                           | 12 |
| 13 | REPOSRPT  | CS     | REPOSBIT | # RETURNS HERE FROM RRIAXIS IN REMODE     | 13 |
| 14 |           |        |          | # REQUESTED DURING REPOSITION.            | 14 |
| 15 |           | MASK   | RADMODES | # REMOVE REPOSITION BIT.                  | 15 |
| 16 |           | TS     | RADMODES |                                           | 16 |
| 17 |           | MASK   | DESIGBIT | # SEE IF SOMEONE IS WAITING TO DESIGNATE. | 17 |
| 18 |           | CCS    | A        |                                           | 18 |
| 19 |           | TCF    | BEGDES   |                                           | 19 |
| 20 |           | CS     | BIT2     | # IF NO FURTHER ANTENNA CONTROL REQUIRED, | 20 |
| 21 |           | EXTEND |          | # REMOVE ERROR COUNTER ENABLE.            | 21 |
| 22 |           | WAND   | CHAN12   |                                           | 22 |
| 23 |           | TCF    | TASKOVER |                                           | 23 |
| 24 |           |        |          |                                           | 24 |
| 25 | SETRRECR  | CAF    | BIT2     | # SET UP RR ERROR COUNTERS                | 25 |
| 26 |           | EXTEND |          |                                           | 26 |
| 27 |           | RAND   | CHAN12   |                                           | 27 |
| 28 |           | CCS    | A        | # DO NOT CLEAR LAST COMMAND IF            | 28 |
| 29 |           | TC     | Q        | # ERROR COUNTERS ARE ENABLED              | 29 |
| 30 |           |        |          |                                           | 30 |
| 31 |           | TS     | LASTYCMD |                                           | 31 |
| 32 |           | TS     | LASTXCMD |                                           | 32 |
| 33 |           | CAF    | BIT2     |                                           | 33 |
| 34 |           | EXTEND |          |                                           | 34 |
| 35 |           | WOR    | CHAN12   | # ENABLE RR CDU ERROR COUNTERS.           | 35 |
| 36 |           | TC     | Q        |                                           | 36 |
| 37 |           |        |          |                                           | 37 |
| 38 |           |        |          |                                           | 38 |
| 39 |           |        |          |                                           | 39 |
| 40 |           |        |          |                                           | 40 |
| 41 |           |        |          |                                           | 41 |
| 42 |           |        |          |                                           | 42 |
| 43 |           |        |          |                                           | 43 |
| 44 |           |        |          |                                           | 44 |
| 45 |           |        |          |                                           | 45 |
| 46 |           |        |          |                                           | 46 |
| 47 |           |        |          |                                           | 47 |
| 48 |           |        |          |                                           | 48 |
| 49 |           |        |          |                                           | 49 |
| 50 |           |        |          |                                           | 50 |
| 51 |           |        |          |                                           | 51 |
| 52 |           |        |          |                                           | 52 |
| 53 |           |        |          |                                           | 53 |
| 54 |           |        |          |                                           | 54 |
| 55 |           |        |          |                                           | 55 |
| 56 |           |        |          |                                           | 56 |
| 57 |           |        |          |                                           | 57 |
| 58 |           |        |          |                                           | 58 |
| 59 |           |        |          |                                           | 59 |
| 60 |           |        |          |                                           | 60 |

# PROGRAM NAME: REMODE

# FUNCTIONAL DESCRIPTION

# REMODE IS THE GENERAL REMODING SUBROUTINE. IT DRIVES THE TRUNNION ANGLE TO 0 DEGREES IF THE CURRENT MODE IS MODE 1, 180 DEGREES FOR MODE 2, THEN DRIVES THE SHAFT ANGLE TO -45 DEGREES, AND FINALLY DRIVES THE TRUNNION ANGLE TO -130 DEGREES, TO PLACE THE RR IN MODE 2, -50 DEGREES FOR MODE 1, BEFORE INITIATING 2-AXIS CONTROL. ALL REMODING IS DONE WITH SINGLE AXIS ROTATIONS (RR1AXIS). INITIALLY THE RR ANTENNA MODE FLAG (RADMODES BIT 12) IS CHECKED. CONTROL IS THEN TRANSFERRED TO RRONLY TO DRIVE THR TRUNNION ANGLE TO 0 DEGREES IF IN MODE 1 OR 180 DEGREES IF IN MODE 2. RRONLY IS THEN CALLED TO DRIVE THE SHAFT ANGLE TO -45 DEGREES. THE RR ANTENNA MODE FLAG (RADMODES BIT 12) IS CHECKED AGAIN. CONTROL IS AGAIN TRANSFERRED TO RRONLY TO DRIVE THE TRUNNION ANGLE TO -130 DEGREES TO PLACE THE RR IN MODE 2 IF CURRENTLY IN MODE 1 OR TO -50 DEGREES IF IN MODE 2 TO PLACE THE RR IN MODE 1. RMODINV IS THEN CALLED TO SET RADMODES BIT 12 TO INDICATE THE NEW RR ANTENNA MODE. THE REMODE FLAG (RADMODES BIT 14) IS REMOVED TO INDICATE THAT REMODING IS COMPLETE. THE PROGRAM THEN EXITS TO STDESIG TO BEGIN 2-AXIS CONTROL.

# CALLIN SEQUENCE:

# FROM BEGDES WHEN REMODE FLAG (RADMODES BIT 14) IS SET. THIS FLAG MAY BE SET IN RRDESSM AND RRDESNB IF RRLIMCHK DETERMINES THAT THE DESIRED ANGLES ARE WITHIN THE LIMITS OF THE OTHER MODE.

# ERASABLE INITIALIZATION REQUIRED:

# RADMODES

# SUBROUTINES CALLED:

# RRONLY, RRSONL, RMODINV (ACTUALLY PART OF)

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: NONE

# EXIT: STDESIG

|        |      |          |                                   |
|--------|------|----------|-----------------------------------|
| REMODE | CAF  | ANTENBIT | # DRIVE TRUNNION TO 0 (180)       |
|        | MASK | RADMODES | # (ERROR COUNTER ALREADY ENABLED) |
|        | CCS  | A        |                                   |
|        | CAF  | BIT15    |                                   |
|        | TC   | RRONLY   |                                   |
|        | CAF  | -45DEGSR |                                   |
|        | TC   | RRONLY   |                                   |

[illegible]

# PROGRAM NAMES: RRONLY, RRONLY

# FUNCTIONAL DESCRIPTION:

# RRONLY AND RRONLY ARE SUBROUTINES FOR DOING SINGLE AXIS  
# RR MANEUVERS FOR REMODE AND REPOSITION. IT DRIVES TO  
# WITHIN 1 DEGREE. INITIALLY, AT RR1AX2, THE REMODE AND REPOSITION  
# FLAGS (RADMODES BITS 14, 11) ARE CHECKED. IF BOTH EXIST,  
# THE PROGRAM EXITS TO REPOSRT (SEE DORREPOS). THIS INDICATES  
# THAT SOMEONE POSSIBLY REQUESTED A DESIGNATE (RADMODES BIT 10)  
# WHICH REQUIRES A REMODE (RADMODES BIT 14) AND THAT A  
# REPOSITION IS IN PROGRESS (RADMODES BIT 11). IF NONE  
# OR ONLY ONE OF THE FLAGS EXIST, REMODE OR REPOSITION, MAGSUB  
# IS CALLED TO SEE IF THE APPROPRIATE ANGLE IS WITHIN 1 DEGREE. IF YES,  
# CONTROL RETURNS TO THE CALLING ROUTINE. IF NOT, CONTROL IS  
# TRANSFERRED TO RROUT FOR SINGLE AXIS MANEUVERS WITH THE OTHER  
# ANGLE SET = 0. FOLLOWING A .5 SECOND WAIT, THE ABOVE PROCEDURE IS  
# REPEATED.

# CALLING SEQUENCE: L-1 CAF \*ANGLE\* (DESIRED ANGLE SCALED PI)  
# L TC RRONLY (TRUNNION ONLY)  
# RRONLY (SHAFT ONLY)  
# RRONLY IS CALLED BY PREPOS29;  
# RRONLY AND RRONLY ARE CALLED BY DORREPOS AND REMODE

# ERASABLE INITIALIZATION REQUIRED:

# C(A) = DESIRED ANGLE, RADMODES

# SUBROUTINES CALLED:

# FIXDELAY, REPOSRT, MAGSUB, RROUT

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: NONE

# EXIT: REPOSRT (REMODE AND REPOSITION FLAGS PRESENT -- RADMODES  
# BITS 14, 11)  
# L+1 (ANGLE WITHIN ONE DEGREE OR RR OUT OF AUTO MODE)

RRONLY TS RDES # DESIRED TRUNNION ANGLE.

CAF ZERO  
TCF RR1AXIS

RRONLY TS RDES # SHAFT COMMANDS ARE UNRESOLVED SINCE THIS  
CAF ONE # ROUTINE ENTERED ONLY WHEN T = 0 OR 180.

RR1AXIS TS RRINDEX  
EXTEND  
QXCH RRRET  
TCF RR1AX2

|          |                           |                              |                                                                                       |
|----------|---------------------------|------------------------------|---------------------------------------------------------------------------------------|
| NXTRR1AX | TC<br>DEC                 | FIXDELAY<br>50               | # 2 SAMPLES PER SECOND.                                                               |
| RR1AX2   | CS<br>MASK                | RADMODES<br>PRIO22           | # IF SOMEONE REQUESTS A DESIGNATE WHICH<br># REQUIRES A REMODE AND A REPOSITION IS IN |
|          | EXTEND<br>BZF             | REPOS RPT                    | # PROGRESS, INTERRUPT IT AND START THE<br># REMODE IMMEDIATELY.                       |
|          | CA<br>EXTEND<br>INDEX     | RDES<br>RRINDEX              |                                                                                       |
|          | MSU<br>TS<br>EXTEND       | CDUT<br>ITEMP1               | # SAVE ERROR SIGNAL.                                                                  |
|          | MP<br>TS<br>CA            | RRSPGAIN<br>L<br>RADMODES    | # TRIES TO NULL .7 OF ERROR OVER NEXT .5                                              |
|          | MASK<br>XCH<br>TC         | AUTOMBIT<br>ITEMP1<br>MAGSUB | # STORE RR-OUT-OF-AUTO-MODE BIT.<br># SEE IF WITHIN ONE DEGREE.                       |
|          | DEC                       | -.00555                      | # SCALED IN HALF-REVS.                                                                |
|          | CCS<br>TC                 | ITEMP1<br>RRRET              | # NO. IF RR OUT OF AUTO MODE, EXIT.<br># RETURN TO CALLER.                            |
|          | CCS<br>TCF<br>XCH<br>DXCH | RRINDEX<br>+2<br>L<br>TANG   | # COMMAND FOR OTHER AXIS IS ZERO.<br># SETTING A TO 0.                                |
|          | TC                        | RROUT                        |                                                                                       |
|          | TCF                       | NXTRR1AX                     | # COME BACK IN .5 SECONDS.                                                            |
| RRSPGAIN | DEC                       | .59062                       | # NULL .7 ERROR IN .5 SEC.                                                            |

# PROGRAM NAME: RROUT

#

# FUNCTIONAL DESCRIPTION:

#

# RROUT RECEIVES RR GYRO COMMANDS IN TANG, TANG +1 IN RR  
# ERROR COUNTER SCALING. RROUT THEN LIMITS THEM AND  
# GENERATES COMMANDS TO THE CDU TO ADJUST THE ERROR COUNTERS  
# TO THE DESIRED VALUES. INITIALLY MAGSUB CHECKS THE MAGNITUDE OF  
# THE COMMAND (SHAFT ON 1ST PASS) TO SEE IF IT IS GREATER THAN  
# 384 PULSES. IF NOT, CONTROL IS TRANSFERRED TO RROUTLIM TO  
# LIMIT THE COMMAND TO +384 OR -384 PULSES. THE DIFFERENCE IS  
# THEN CALCULATED BETWEEN THE DESIRED STATE AND THE PRESENT STATE OF  
# THE ERROR COUNTER AS RECORDED IN LASTYCMD AND LASTXCMD.  
# THE RESULT IS STORED IN OPTXCMD (1ST PASS) AND OPTYCMD (2ND  
# PASS). FOLLOWING THE SECOND PASS, FOR THE TRUNNION COMMAND, THE  
# OCDUT AND OCDUS ERROR COUNTER DRIVE BITS (CHAN 14 BITS 12, 11)  
# ARE SET. THIS PROGRAM THEN EXITS TO THE CALLING PROGRAM.

# CALLING SEQUENCE:

#

# L TC RROUT (WITH RUPT INHIBITED) RROUT IS CALLED BY  
# RRONLY, RRONLY, AND DODES

#

# ERASABLE INITIALIZATION REQUIRED:

#

# TANG, TANG +1 (DESIRED COMMANDS), LASTYCMD, LASTXCMD  
# (1ST PASS = 0), RR ERROR COUNTER ENAGLE SET (CHAN 12 BIT 2).

#

# SUBROUTINES CALLED:

#

# MAGSUB

#

# JOBS OR TASKS INITIATED:

#

# NONE

#

# ALARMS: NONE

#

# EXIT: L+1 (ALWAYS)

|          |       |          |                                          |
|----------|-------|----------|------------------------------------------|
| RROUT    | LXCH  | Q        | # SAVE RETURN                            |
|          | CAF   | ONE      | # LOOP TWICE.                            |
| RROUT2   | TS    | ITEMP2   |                                          |
|          | INDEX | A        |                                          |
|          | CA    | TANG     |                                          |
|          | TS    | ITEMP1   | # SAVE SIGN COMMAND FOR LIMITING.        |
|          |       |          |                                          |
|          | TC    | MAGSUB   | # SEE IF WITHIN LIMITS.                  |
| -RRLIMIT | DEC   | -384     |                                          |
|          | TCF   | RROUTLIM | # LIMIT COMMAND TO MAG OF 384.           |
|          |       |          |                                          |
| SETRRCTR | CA    | ITEMP1   | # COUNT OUT DIFFERENCE BETWEEN DESIRED   |
|          | INDEX | ITEMP2   | # STATE AND PRESENT STATE AS RECORDED IN |
|          | XCH   | LASTYCMD | # LASTYCMD AND LASTXCMD                  |
|          | COM   |          |                                          |



|    |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |
| 8  |
| 9  |
| 10 |
| 11 |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 |
| 17 |
| 18 |
| 19 |
| 20 |
| 21 |
| 22 |
| 23 |
| 24 |
| 25 |
| 26 |
| 27 |

# ROUTINE TO ZERO THE RR CDUS AND DETERMINE THE ANTENNA MODE.

RRZERO CAF BIT11+1 # SEE IF MONITOR REPOSITION OR NOT IN AUTO  
MASK RADMODES # IF SO, DON'T RE-ZERO CDUS.  
CCS A

TCF RADNOOP # (IMMEDIATE TASK TO RGOODEND).

INHINT

CS RCDOUBIT # SET FLAG TO SHOW ZEROING IN PROGRESS.  
MASK RADMODES  
AD RCDOUBIT

TS RADMODES

CAF ONE  
TC WAITLIST  
EBANK= LOSCOUNT  
2CADR RRZ2

CS RADMODES # SEE IF IN AUTO MODE.  
MASK AUTOMBIT

CCS A  
TCF ROADBACK  
TC ALARM # AUTO DISCRETE NOT PRESENT -- TRYING

ROADBACK OCT 510  
RELINT  
TCF SWRETURN

RRZ2 TC RRZEROSB # COMMON TO TURNON AND RRZERO.  
TCF ENDRADAR

BIT11+1 OCT 02001

1412THE

# PROGRAM NAME: RRDESSM

# FUNCTIONAL DESCRIPTION:

# THIS INTERPRETIVE ROUTINE WILL DESIGNATE, IF DESIRED ANGLES ARE  
# WITHIN THE LIMITS OF EITHER MODE, TO A LINE-OF-SIGHT (LOS) VECTOR  
# (HALF-UNIT) KNOWN WITH RESPECT TO THE STABLE MEMBER PRESENT  
# ORIENTATION. INITIALLY THE IMU CDU'S ARE READ AND CONTROL  
# TRANSFERRED TO SMNB TO TRANSFORM THE LOS VECTOR FROM STABLE  
# MEMBER TO NAVIGATION BASE COORDINATES (SEE STG MEMO 699)  
# RRANGLES IS THEN CALLED TO CALCULATE THE RR GIMBAL ANGLES,  
# TRUNNION AND SHAFT, FOR BOT THE PRESENT AND ALTERNATE MODE.  
# RRLIMCHK IS CALLED TO SEE IF THE ANGLES CALCULATED FOR THE  
# PRESENT MODE ARE WITHIN LIMITS. IF WITHIN LIMITS, THE RETUREN  
# LOCATION IS INCREMENTED, INASMUCH AS NO VEHICLE MANEUVER IS  
# REQUIRED, BEFORE EXITING TO STARTDES. IF NOT WITHIN THE LIMITS OF THE  
# CURRENT MODE, TRYWS IS CALLED. FOLLOWING INVERTING OF THE RR  
# ANTENNA MODE FLAG (RADMODES BIT 12), RRLIMCHK IS CALLED  
# TO SEE IF THE ANGLES CALCULATED FOR THE ALTERNATE MODE ARE WITHIN  
# LIMITS. IF YES, THE RR ANTENNA MODE FLAG IS AGAIN INVERTED,  
# THE REMODE FLAG (RADMODES BIT 14) SET, AND THE RETURN LOCATION  
# INCREMENTED, TO INDICATE NO VEHICLE MANEUVER IS REQUIRED, BEFORE  
# EXITING TO STARTDES. IF THESE ANGLES ARE NOT WITHIN LIMITS  
# OF THE ALTERNATE MODE, THE RR ANTENNA MODE FLAG (RADMODES  
# BIT 12) IS INVERTED BEFORE RETURNING DIRECTLY TO THE CALLING PROGRAM  
# TO INDICATE THAT A VEHICLE MANEUVER IS REQUIRED.

# CALLING SEQUENCE:

|   |     |        |          |                                          |
|---|-----|--------|----------|------------------------------------------|
| # | L   | STCALL | RRTARGET | (LOS HALF-UNIT VECTOR IN SM COORDINATES) |
| # | L+1 | RRDESM |          |                                          |
| # | L+2 | BASIC  |          | (VEHICLE MANEUVER REQUIRED)              |
| # | L+3 | BASIC  |          | (NO VEHICLE MANEUVER REQUIRED)           |

# ERASABLE INITIALIZATION REQUIRED:

# RRTARGET, RADMODES

# SUBROUTINES CALLED:

# READCDUS, SMNB, RRANGLES, RRLIMCHK, TRYWS (ACTUALLY  
# PART OF), RMODINV

# JOBS OR TASKS INITIATED:

# NONE

# ALARMS: NONE

# EXIT: L+2 (NEITHER SET OF ANGLES ARE WITHIN LIMITS OF RELATED MODE)  
# STARTDES (DESIGNATE POSSIBLE AT PRESENT VEHICLES ATTITUDE -- RETURNS  
# TO L+3 FROM STARTDES)RRDESSM STQ CLEAR  
DESRET

|    |          |        |          |                                                  |
|----|----------|--------|----------|--------------------------------------------------|
| 1  |          |        | RRNBSW   |                                                  |
| 2  |          | CALL   |          | # COMPUTES SINES AND COSINES, ORDER Y Z X        |
| 3  |          |        |          |                                                  |
| 4  |          | VLOAD  | CDUTRIG  |                                                  |
| 5  |          |        | CALL     | # LOAD VECTOR AND CALL TRANSFORMATION            |
| 6  |          |        | RRTARGET |                                                  |
| 7  |          |        | *SMNB*   |                                                  |
| 8  |          |        |          |                                                  |
| 9  |          | CALL   |          | # GET RR GIMBAL ANGLES IN PRESENT AND            |
| 10 |          |        | RRANGLES | # ALTERNATE MODE.                                |
| 11 |          | EXIT   |          |                                                  |
| 12 |          |        |          |                                                  |
| 13 |          | INHINT |          |                                                  |
| 14 |          | TC     | RRLIMCHK |                                                  |
| 15 |          | ADRES  | MODEA    | # CONFIGURATION FOR CURRENT MODE.                |
| 16 |          | TC     | +3       | # NOT IN CURRENT MODE                            |
| 17 | OKDESSM  | INCR   | DESRET   | # INCREMENT SAYS NO VEHICLE MANEUVER REQ.        |
| 18 |          | TC     | STARTDES | # SHOW DESIGNATE REQUIRED                        |
| 19 |          | CS     | FLAGWRD8 |                                                  |
| 20 |          | MASK   | SURFFBIT | # CHECK IF ON LUNAR SURFACE (SURFFLAG=P22F)      |
| 21 |          | EXTEND |          |                                                  |
| 22 |          | BZF    | NORDSTAL | # BRANCH -- YES -- CANNOT DESIGNATE IN MODE 2    |
| 23 |          | TC     | TRYSW    |                                                  |
| 24 |          |        |          |                                                  |
| 25 | LUNDESCH | CS     | FLAGWRD8 | # OVERFLOW RETURN FROM RRANGLES                  |
| 26 |          | MASK   | SURFFBIT | # CHECK IF ON LUNAR SURFACE                      |
| 27 |          | EXTEND |          |                                                  |
| 28 |          | BZF    | NORDSTAL | # BRANCH -- YES -- RETURN TO CALLER -- ALARM 527 |
| 29 |          | CA     | STATE    |                                                  |
| 30 |          | MASK   | RNDVZBIT |                                                  |
| 31 |          | CCS    | A        | # TEST RNDVZFLG                                  |
| 32 |          | TC     | NODESSM  | # NOT ON MOON -- CALL FOR ATTITUDE MANEUVER      |
| 33 |          | TCF    | ENDOFJOB | # ... BUT NOT IN R29.                            |
| 34 |          |        |          |                                                  |
| 35 |          |        |          |                                                  |
| 36 |          |        |          |                                                  |
| 37 |          |        |          |                                                  |
| 38 |          |        |          |                                                  |
| 39 |          |        |          |                                                  |
| 40 |          |        |          |                                                  |
| 41 |          |        |          |                                                  |
| 42 |          |        |          |                                                  |
| 43 |          |        |          |                                                  |
| 44 |          |        |          |                                                  |
| 45 |          |        |          |                                                  |
| 46 |          |        |          |                                                  |
| 47 |          |        |          |                                                  |
| 48 |          |        |          |                                                  |
| 49 |          |        |          |                                                  |
| 50 |          |        |          |                                                  |
| 51 |          |        |          |                                                  |
| 52 |          |        |          |                                                  |
| 53 |          |        |          |                                                  |
| 54 |          |        |          |                                                  |
| 55 |          |        |          |                                                  |
| 56 |          |        |          |                                                  |
| 57 |          |        |          |                                                  |
| 58 |          |        |          |                                                  |
| 59 |          |        |          |                                                  |
| 60 |          |        |          |                                                  |

# PROGRAM NAME: STARTDES

# FUNCTIONAL DESCRIPTION:

# STARTDES IS ENTERED WHEN WE ARE READY TO BEGIN DESIGNATION.  
# BIT 14 OF RADMODES IS ALREADY SET IF A REMODE IS REQUIRED.  
# AT THIS TIME, THE RR ANTENNA MAY BE IN A REPOSITON  
# OPERATION. IN THIS CASE, IF A REMODE IS REQUIRED IT MAY HAVE  
# ALREADY BEGUN BUT IN ANY CASE THE REPOSITION WILL BE INTERRUPTED.  
# OTHERWISE, THE REPOSITION WILL BE COMPLETED BEFORE 2-AXIS  
# DESIGNATION BEGINS. INITIALLY DESCOUNT IS SET = 60 TO INDICATE  
# THAT 30 SECONDS WILL BE ALLOWED FOR THE RR DATA GOOD INBIT  
# (CHAN 33 BIT 4) IF LOCK-ON IS DESIRED (STATE BIT 5). BIT 10  
# OF RADMODES IS SET TO SHOW THAT A DESIGNATE IS REQUIRED.  
# THE REPOSITON FLAG (RADMODES BIT 11) IS CHECKED. IF SET,  
# THE PROGRAM EXITS TO L+3 OF THE CALLING PROGRAM (SEE RRDESSM  
# AND RRDESNB). THE PROGRAM WILL BEGIN DESIGNATING TO THE DESIRED  
# ANGLES FOLLOWING THE REPOSITON OR REMODE IF ONE WAS  
# REQUESTED. IF THE REPOSITON FLAG IS NOT SET, SETRRECR IS CALLED  
# WITH SETS THE RR ERROR COUNTER ENABLE BIT (CHAN 12 BIT 2)  
# AND SETS LASTYCMD AND LASTXCMD = 0 TO INDICATE THE  
# DIFFERENCE BETWEEN THE PRESENT AND DESIRED STATE OF THE ERROR  
# COUNTERS. A 20 MILLISECOND WAITLIST CALL IS SET FOR BEGDES  
# AFTER WHICH THE PROGRAM EXITS TO L+3 OF THE CALLING PROGRAM.

# CALLING SEQUENCE:

# FROM RRDESSM AND RRDESNB WHEN ANGLES WITHIN LIMITS.

# ERASABLE INITIALIZATION REQUIRED:

# RADMODES, (SEE DODES)

# SUBROUTINES CALLED

# SETRRECR, WAITLIST

# JOBS OR TASKS INITIATED:

# BEGDES

# ALARMS: NONE

# EXIT: L+3 OF CALLING PROGRAM (SEE RRDESSM)

# L+2 OF CALLING PROGRAM (SEE RRDESNB)

|          |      |          |                                     |
|----------|------|----------|-------------------------------------|
| STARTDES | INCR | DESRET   |                                     |
|          | CS   | RADMODES |                                     |
|          | MASK | DESIGBIT |                                     |
|          | ADS  | RADMODES |                                     |
|          | MASK | REPOSBIT | # SEE IF REPOSITIONING IN PROGRESS. |
|          | CCS  | A        |                                     |
|          | TCF  | DESRETRN | # ECTR ALREADY SET UP.              |
|          | TC   | SETRRECR | # SET UP ERROR COUNTERS.            |

[illegible]



|    |                                                      |       |          |                             |    |
|----|------------------------------------------------------|-------|----------|-----------------------------|----|
| 1  | # SEE IF RRDESSM CAN BE ACCOMPLISHED AFTER A REMODE. |       |          |                             | 1  |
| 2  |                                                      |       |          |                             | 2  |
| 3  |                                                      |       |          |                             | 3  |
| 4  | TRYSWS                                               | TC    | RMODINV  | # (NOTE RUPT INHIBIT)       | 4  |
| 5  |                                                      | TC    | RRLIMCHK | # TRY DIFFERENT MODE.       | 5  |
| 6  |                                                      | ADRES | MODEB    |                             | 6  |
| 7  |                                                      | TCF   | NODESSM  | # VEHICLE MANEUVER REQUIRED | 7  |
| 8  |                                                      |       |          |                             | 8  |
| 9  |                                                      | TC    | RMODINV  | # RESET BIT12               | 9  |
| 10 |                                                      | CAF   | REMODBIT | # SET FLAG FOR REMODE.      | 10 |
| 11 |                                                      | ADS   | RADMODES |                             | 11 |
| 12 |                                                      |       |          |                             | 12 |
| 13 |                                                      | TCF   | OKDESSM  |                             | 13 |
| 14 |                                                      |       |          |                             | 14 |
| 15 | NODESSM                                              | TC    | RMODINV  | # RE-INVERT MODE AND RETURN | 15 |
| 16 |                                                      | INCR  | DESRET   | # TO CALLER +2              | 16 |
| 17 |                                                      | TCF   | NORDSTAL |                             | 17 |
| 18 |                                                      |       |          |                             | 18 |
| 19 | MAXTRYS                                              | DEC   | 60       |                             | 19 |
| 20 |                                                      |       |          |                             | 20 |
| 21 |                                                      |       |          |                             | 21 |
| 22 |                                                      |       |          |                             | 22 |
| 23 |                                                      |       |          |                             | 23 |
| 24 |                                                      |       |          |                             | 24 |
| 25 |                                                      |       |          |                             | 25 |
| 26 |                                                      |       |          |                             | 26 |
| 27 |                                                      |       |          |                             | 27 |
| 28 |                                                      |       |          |                             | 28 |
| 29 |                                                      |       |          |                             | 29 |
| 30 |                                                      |       |          |                             | 30 |
| 31 |                                                      |       |          |                             | 31 |
| 32 |                                                      |       |          |                             | 32 |
| 33 |                                                      |       |          |                             | 33 |
| 34 |                                                      |       |          |                             | 34 |
| 35 |                                                      |       |          |                             | 35 |
| 36 |                                                      |       |          |                             | 36 |
| 37 |                                                      |       |          |                             | 37 |
| 38 |                                                      |       |          |                             | 38 |
| 39 |                                                      |       |          |                             | 39 |
| 40 |                                                      |       |          |                             | 40 |
| 41 |                                                      |       |          |                             | 41 |
| 42 |                                                      |       |          |                             | 42 |
| 43 |                                                      |       |          |                             | 43 |
| 44 |                                                      |       |          |                             | 44 |
| 45 |                                                      |       |          |                             | 45 |
| 46 |                                                      |       |          |                             | 46 |
| 47 |                                                      |       |          |                             | 47 |
| 48 |                                                      |       |          |                             | 48 |
| 49 |                                                      |       |          |                             | 49 |
| 50 |                                                      |       |          |                             | 50 |
| 51 |                                                      |       |          |                             | 51 |
| 52 |                                                      |       |          |                             | 52 |
| 53 |                                                      |       |          |                             | 53 |
| 54 |                                                      |       |          |                             | 54 |
| 55 |                                                      |       |          |                             | 55 |
| 56 |                                                      |       |          |                             | 56 |
| 57 |                                                      |       |          |                             | 57 |
| 58 |                                                      |       |          |                             | 58 |
| 59 |                                                      |       |          |                             | 59 |
| 60 |                                                      |       |          |                             | 60 |

# DESIGNATE TO SPECIFIC RR GIMBAL ANGLES (INDEPENDENT OF VEHICLE MOTION). ENTER WITH DESIRED ANGLES IN  
# TANG AND TANG +1.

|         |                    |                           |                                                                          |
|---------|--------------------|---------------------------|--------------------------------------------------------------------------|
| RRDESNB | TC<br>TS           | MAKECADR<br>DESRET        |                                                                          |
|         | TC<br>ADRES        | DOWNFLAG<br>LOSCMFLG      | # RESET FLAG TO PREVENT DODES FROM GOING<br># BACK TO R21                |
|         | CA<br>TS<br>INHINT | MAXTRYS<br>DESCOUNT       | # SET TIME LIMIT COUNTER<br># FOR DESIGNATE<br># SEE IF CURRENT MODE OK. |
|         | TC<br>ADRES<br>TCF | RRLIMNB<br>TANG<br>TRYSWN | # DO SPECIAL V41 LIMIT CHECK<br># SEE IF IN OTHER MODE.                  |

|         |                   |                           |                         |
|---------|-------------------|---------------------------|-------------------------|
| OKDESNB | RELINT<br>EXTEND  |                           |                         |
|         | DCA<br>DXCH<br>TC | TANG<br>TANGNB<br>INTPRET |                         |
|         | CALL              |                           | # GET LOS IN NB COORDS. |
|         |                   | RRNB                      |                         |
|         | STORE             | RRTARGET                  |                         |
|         | SET               | EXIT<br>RRNBSW            |                         |

|        |                         |                                            |                              |
|--------|-------------------------|--------------------------------------------|------------------------------|
|        | INHINT                  |                                            |                              |
| TRYSWN | TCF                     | STARTDES +1                                |                              |
|        | TC                      | RMODINV                                    | # SEE IF OTHER MODE WILL DO. |
|        | TC                      | RRLIMNB                                    | # DO SPECIAL V41 LIMIT CHECK |
|        | ADRES<br>TCF            | TANG<br>NODESNB                            | # NOT POSSIBLE.              |
|        | TC<br>CAF<br>ADS<br>TCF | RMODINV<br>REMODBIT<br>RADMODES<br>OKDESNB | # CALL FOR REMODE.           |

|         |           |                 |                      |
|---------|-----------|-----------------|----------------------|
| NODESNB | TC        | RMODINV         | # REINVERT MODE BIT. |
|         | TC        | ALARM           | # BAD INPUT ANGLES.  |
|         | OCT<br>TC | 502<br>CLRADMOD |                      |
|         | TC        | ENDOFJOB        | # AVOID 503 ALARM.   |

|         |                |   |                                         |
|---------|----------------|---|-----------------------------------------|
| RRLIMNB | INDEX          | Q | # THIS ROUTINE IS IDENTICAL TO RRLIMCHK |
|         | CAF            | 0 | # EXCEPT THAT THE MODE 1 SHAFT LOWER    |
|         | INCR<br>EXTEND | Q | # LIMIT IS -85 INSTEAD OF -70 DEGREES   |



|    |          |        |          |                                          |  |    |
|----|----------|--------|----------|------------------------------------------|--|----|
| 1  |          |        |          |                                          |  | 1  |
| 2  |          | INDEX  | A        | # READ GIMBAL ANGLES INTO ITEMP STORAGE  |  | 2  |
| 3  |          | DCA    | 0        |                                          |  | 3  |
| 4  |          | DXCH   | ITEMP1   |                                          |  | 4  |
| 5  |          | LXCH   | Q        | # L(CALLER +2) TO L                      |  | 5  |
| 6  |          |        |          |                                          |  | 6  |
| 7  |          | CAF    | ANTENBIT | # SEE WHICH MODE RR IS IN                |  | 7  |
| 8  |          | MASK   | RADMODES |                                          |  | 8  |
| 9  |          | CCS    | A        |                                          |  | 9  |
| 10 |          | TCF    | MODE2CHK | # MODE 2 CAN USE RRLIMCHK CODING         |  | 10 |
| 11 |          | CA     | ITEMP1   |                                          |  | 11 |
| 12 |          | TC     | MAGSUB   | # MODE 1 IS DEFINED AS                   |  | 12 |
| 13 |          | DEC    | -.30555  | # 1 ABS(T) L 55 DEGS                     |  | 13 |
| 14 |          | TC     | L        | # 2 SHAFT LIMITS AT +59, -85 DEGS        |  | 14 |
| 15 |          |        |          |                                          |  | 15 |
| 16 |          | CA     | ITEMP2   | # LOAD SHAFT ANGLE                       |  | 16 |
| 17 |          | EXTEND |          |                                          |  | 17 |
| 18 |          | BZMF   | NEGSHAFT | # IF NEGATIVE SHAFT ANGLE, ADD 20.5 DEGS |  | 18 |
| 19 |          | AD     | 5.5DEGS  |                                          |  | 19 |
| 20 | SHAFTLIM | TC     | MAGSUB   |                                          |  | 20 |
| 21 |          | DEC    | -.35833  | # 64.5 DEGREES                           |  | 21 |
| 22 |          | TC     | L        | # NOT IN LIMITS                          |  | 22 |
| 23 |          | TC     | RRLIMOK  | # IN LIMITS                              |  | 23 |
| 24 | NEGSHAFT | AD     | 20.5DEGS | # MAKE NEGATIVE SHAFT LIMIT -85 DEGREES  |  | 24 |
| 25 |          | TCF    | SHAFTLIM |                                          |  | 25 |
| 26 |          |        |          |                                          |  | 26 |
| 27 | 20.5DEGS | DEC    | .11389   |                                          |  | 27 |
| 28 |          |        |          |                                          |  | 28 |
| 29 |          |        |          |                                          |  | 29 |
| 30 |          |        |          |                                          |  | 30 |
| 31 |          |        |          |                                          |  | 31 |
| 32 |          |        |          |                                          |  | 32 |
| 33 |          |        |          |                                          |  | 33 |
| 34 |          |        |          |                                          |  | 34 |
| 35 |          |        |          |                                          |  | 35 |
| 36 |          |        |          |                                          |  | 36 |
| 37 |          |        |          |                                          |  | 37 |
| 38 |          |        |          |                                          |  | 38 |
| 39 |          |        |          |                                          |  | 39 |
| 40 |          |        |          |                                          |  | 40 |
| 41 |          |        |          |                                          |  | 41 |
| 42 |          |        |          |                                          |  | 42 |
| 43 |          |        |          |                                          |  | 43 |
| 44 |          |        |          |                                          |  | 44 |
| 45 |          |        |          |                                          |  | 45 |
| 46 |          |        |          |                                          |  | 46 |
| 47 |          |        |          |                                          |  | 47 |
| 48 |          |        |          |                                          |  | 48 |
| 49 |          |        |          |                                          |  | 49 |
| 50 |          |        |          |                                          |  | 50 |
| 51 |          |        |          |                                          |  | 51 |
| 52 |          |        |          |                                          |  | 52 |
| 53 |          |        |          |                                          |  | 53 |
| 54 |          |        |          |                                          |  | 54 |
| 55 |          |        |          |                                          |  | 55 |
| 56 |          |        |          |                                          |  | 56 |
| 57 |          |        |          |                                          |  | 57 |
| 58 |          |        |          |                                          |  | 58 |
| 59 |          |        |          |                                          |  | 59 |
| 60 |          |        |          |                                          |  | 60 |

# PROGRAM NAME: BEGDES

# FUNCTIONAL DESCRIPTION:

# BEGDES CHECKS VARIOUS DESIGNATE REQUESTS AND REQUESTS THE  
# ACTUAL RR DESIGNATION. INITIALLY A CHECK IS MADE TO SEE IF A  
# REMODE (RADMODES BIT 14) IS REQUESTED OR IN PROGRESS. IF SO,  
# CONTROL IS TRANSFERRED TO STDESIG AFTER ROUTINE REMODE IS  
# EXECUTED. IF NO REMODE, STDESIG IS IMMEDIATELY CALLED WHERE  
# FIRST THE REPOSITION FLAG (RADMODES BIT 11) IS CHECKED. IF  
# PRESENT, THE DESIGNATE FLAG (RADMODES BIT 10) IS REMOVED  
# AFTER WHICH THE PROGRAM EXITS TO RDBADEND. IF THE REPOSITION  
# FLAG IS NOT PRESET, THE CONTINUOUS DESIGNATE FLAG (RADMODES  
# BIT 15) IS CHECKED. IF PRESENT, AN EXECUTIVE CALL IS IMMEDIATELY  
# MADE FOR DODES AFTER WHICH A .5 SECOND WAIT IS INITIATED BEFORE  
# REPEATING AT STDESIG. IF THE RR SEARCH ROUTINE (LRS24.1) IS DESIGNATING  
# TO A NEW POINT (NEWPTFLG SET) THE CURRENT DESIGNATE TASK IS TERMINATED.  
# IF CONTINUOUS DESIGNATE IS NOT WANTED, THE DESIGNATE FLAG (RADMODES  
# BIT 10) IS CHECKED. IF NOT PRESENT, THE PROGRAM EXITS TO ENDRADAR TO  
# CHECK RR CDU FAIL BEFORE RETURNING TO THE CALLING PROGRAM. IF DESIGNATE  
# IS STILL REQUIRED, DESCOUNT IS CHECKED TO SEE IF THE 30 SECONDS HAS  
# EXPIRED BEFORE RECEIVING THE RR DATA GOOD (CHAN 33 BIT 4)  
# SIGNAL. IF OUT OF TIME, PROGRAM ALARM 00503 IS REQUESTED, THE  
# RR AUTO TRACKER ENABLE AND RR ERROR COUNTER ENABLE  
# (CHAN 12 BITS 14,2) BITS REMOVED, AND THE DESIGNATE FLAG  
# (RADMODES BIT 10) REMOVED BEFORE EEXITING TO RDBADEND. IF  
# TIME HAS NOT EXPIRED, DESCOUNT IS DECREMENTED, THE  
# EXECUTIVE CALL MADE FOR DODES, AND A .5 SECOND WAIT INITIATED  
# BEFORE REPEATING THIS PROCEDURE AT STDESIG.

# CALLING SEQUENCE:

# WAITLIST CALL FROM STARTDES  
# TCF BEGDES FROM DORREPOS  
# TC STDESIG RETURNING, FROM REMODE

# ERASABLE INITIALIZATION REQUIRED:

# DESCOUNT, FINDVAC

# JOBS OR TASKS INITIATED: DODES

# ALARMS: PROGRAM ALARM 00503 (30 SECONDS HAVE EXPIRED) WITH NO RR DATA  
# GOOD (CHAN 33 BIT 4) RECEIVED WHEN LOCK-ON (STATE BIT 5) WAS REQUESTED.

# EXIT: TASKOVER (SEARCH PATTERN DESIGNATING TO NEW POINT)  
# ENDRADAR (NO DESIGNATE -- RADMODES BIT 10)  
# RDBADEND (REPOSITION OR 30 SECONDS EXPIRED)

BEGDES CS RADMODES

|          |        |            |                                            |
|----------|--------|------------|--------------------------------------------|
|          | MASK   | REMODBIT   |                                            |
|          | CCS    | A          |                                            |
|          | TC     | STDESIG    |                                            |
|          | TC     | REMODE     |                                            |
| DESLOOP  | TC     | FIXDELAY   | # 2 SAMPLES PER SECOND.                    |
|          | DEC    | 50         |                                            |
| STDESIG  | CAF    | REPOSBIT   |                                            |
|          | MASK   | RADMODES   | # SEE IF GIMBAL LIMIT MONITOR HAS FOUND US |
|          | CCS    | A          | # OUT OF BOUNDS. IF SO, THIS BIT SHOWS A   |
|          | TCF    | BADDES     | # REPOSITION TO BE IN PROGRESS.            |
|          | CCS    | RADMODES   | # SEE IF CONTINUOUS DESIGNATE WANTED.      |
|          | TCF    | +3         | # IF SO, DON'T CHECK BIT 10 TO SEE IF IN   |
|          | TCF    | +2         | # LIMITS BUT GO RIGHT TO FINDVAC ENTRY.    |
|          | TCF    | MOREDES +1 |                                            |
|          | CS     | RADMODES   | # IF NON-CONTINUOUS, SEE IF END OF         |
|          | MASK   | DESIGBIT   | # PROBLEM (DATA GOOD IF LOCK-ON WANTED OR  |
|          | CCS    | A          | # WITHIN LIMITS IF NOT). IF SO, EXIT AFTER |
|          | TCF    | ENDRADAR   | # CHECKING RR CDU FAIL.                    |
| STDESIG1 | CCS    | DESCOUNT   | # SEE IF THE TIME LIMIT HAS EXPIRED        |
|          | TCF    | MOREDES    |                                            |
|          | CS     | B14+B2     | # IF OUT OF TIME, REMOVE ECR ENABLE + TRKR |
|          | EXTEND |            |                                            |
|          | WAND   | CHAN12     |                                            |
| BADDES   | CS     | DESIGBIT   | # REMOVE DESIGNATE FLAG                    |
|          | MASK   | RADMODES   |                                            |
|          | TS     | RADMODES   |                                            |
|          | TCF    | RDBADEND   |                                            |
| MOREDES  | TS     | DESCOUNT   |                                            |
|          | CAF    | PRI026     | # UPDATE GYRO TORQUE COMMANDS.             |
|          | TC     | FINDVAC    |                                            |
|          | EBANK= | LOSCOUNT   |                                            |
|          | 2CADR  | DODES      |                                            |
|          | TCF    | DESLOOP    |                                            |
| B14+B2   | OCT    | 20002      |                                            |

# PROGRAM NAME: DODES

# FUNCTIONAL DESCRIPTION:

# DODES CALCULATES AND REQUESTS ISSUANCE OF RR GYRO TORQUE  
# COMMANDS. INITIALLY THE CURRENT RR CDU ANGLES ARE STORED AND  
# THE LOS HALF-UNIT VECTOR TRANSFORMED FROM STABLE MEMBER TO  
# NAVIGATION BASE COORDINATES VIA SMNB IF NECESSARY. THE  
# SHAFT AND TRUNNION COMMANDS ARE THEN CALCULATED AS FOLLOWS:  
# + SHAFT = LOS . (COS(S), 0, -SIN(S)) (DOT PRODUCT)  
# - TRUNNION = LOS . (SIN(T)SIN(S), COS(T), SIN(T)COS(S))  
# THE SIGN OF THE SHAFT COMMAND IS THEN REVERSED IF IN MODE 2  
# (RADMODES BIT 12) BECAUSE A RELAY IN THE RR REVERSES THE  
# POLARITY OF THE COMMAND. AT RRSCALUP EACH COMMAND IS  
# SCALED AND IF EITHER, OR BOTH, OF THE COMMANDS IS GREATER THAN  
# .5 DEGREES, MPAC +1 IS SET POSITIVE. IF A CONTINUOUS DESIGNATE  
# (RADMODES BIT 15) IS DESIRED AND THE SEARCH ROUTINE IS NOT OPERATING,  
# THE RR AUTO TRACKER ENABLE BIT (CHAN 12 BIT 14) IS CLEARED AND RROUT  
# CALLED TO PUT OUT THE COMMANDS PROVIDED NO REPOSITION (RADMODES BIT 11)  
# IS IN PROGRESS. IF A CONTINUOUS DESIGNATE AND THE SEARCH ROUTINE IS  
# OPERATING (SRCHOPT FLAT SET) THE TRACK ENABLE IS NOT CLEARED. IF NO  
# CONTINUOUS DESIGNATE AND BOTH COMMANDS ARE NOT LESS THAN .5 DEGREES AS  
# INDICATED BY MPAC +1, THE RR AUTO TRACKER ENABLE BIT (CHAN 12 BIT 14) IS  
# CLEARED AND RROUT CALLED TO PUT OUT THE COMMANDS PROVIDED NO REPOSITON  
# (RADMODES BIT 11) IS IN PROGRESS. IF BOTH COMMANDS ARE LESS THAN .5  
# DEGREES AS INDICATED BY MPAC+1, THE RR AUTO TRACKER ENABLE BIT  
# (CHAN 12 BIT 14) IS CLEARED AND RROUT CALLED TO PUT OUT THE  
# COMMANDS PROVIDED NO REPOSITION (RADMODES BIT 11) IS IN  
# PROGRESS. IF BOTH COMMANDS ARE LESS THAN .5 DEGREES, THE  
# LOCK-ON FLAG (STATE BIT 5) IS CHECKED. IF NOT PRESETN, THE  
# DESIGNATE FLAG (RADMODES BIT 10) IS CLEARED, AND ENDOFJOB  
# CALLED. IF LOCK-ON IS DESIRED, THE RR AUTO TRACKER (CHAN 12  
# BIT 14) IS ENABLED FOLLOWED BY A CHECK OF THE RECEIPT OF THE  
# RR DATA GOOD (CHAN 33 BIT 4) SIGNAL. IF RR DATA GOOD  
# PRESENT, THE DESIGNATE FLAG (RADMODES BIT 10) IS CLEARED,  
# THE RR ERROR COUNTER ENABLE BIT (CHAN 12 BIT 2) IS CLEARED,  
# AND ENDOFJOB CALLED. IF RR DATA GOOD IS NOT PRESENT, RROUT  
# IS CALLED TO PUT OUT THE COMMANDS PROVIDED NO REPOSITION  
# (RADMODES BIT 11) IS IN PROGRESS AFTER WHICH THE JOB IS TERMINATED  
# VIA ENDOFJOB.

# CALLING SEQUENCE:

# EXECUTIVE CALL EVERY .5 SECONDS FROM BEGDES.

# ERASABLE INITIALIZATION REQUIRED:

# RRTARGET (HALF-UNIT LOS VECTOR IN EITHER SM OR NB COORDINATES),  
# LOKONSW (STATE BIT 5), RRNSW (STATE BIT 6), RADMODES

# SUBROUTINES CALLED:

# READCDUS, SMNB, CDULOGIC, MAGSUB, RROUT

#  
# JOBS OR TASKS INITIATED:

#  
# NONE

#  
# ALARMS: NONE

#  
# EXIT: ENDOFJOB (ALWAYS)

DODES

EXTEND

DCA CDUT

DXCH TANG

TC INTPRET

SETPD VLOAD

0

BON RRTARGET

VXSC

RRNBSW

DONBRD

# TARGET IN NAV-BASE COORDINATES

MLOSV

# MULTIPLY UNIT LOS BY MAGNITUDE

VSL1 PDVL

LOSVEL

VXSC VAD

# ADD ONE SECOND RELATIVE VELOCITY TO LOS

MCTOMS

UNIT CALL

CDUTRIG

CALL

\*SMNB\*

DONBRD

STODL

32D

RTB

TANG +1

PUSH

# SHAFT COMMAND = V(32D).(COS(S), 0,

CDULOGIC

# -SIN(S)).

SIN

PDDL

# SIN(S) TO 0 AND COS(S) TO 2.

COS

PUSH

DMP

PDDL

32D

36D

DMP

BDSU

0

STADR

STORE

TANG +1

# SHAFT COMMAND

SLOAD

RTB

TANG

CDULOGIC

PUSH

COS

# COS(T) TO 4.

PDDL

SIN

PUSH

DMP

# SIN(T) TO 6.

2

|      |      |   |                   |                |
|------|------|---|-------------------|----------------|
| SL1  | PDDL | # | DEFINE VECTOR U = | [SIN(T)SIN(S)] |
|      | 4    | # |                   | [ COS(T) ]     |
| PDDL | DMP  | # |                   | [SIN(T)COS(S)] |

|     |      |                                           |
|-----|------|-------------------------------------------|
| SL1 | VDEF |                                           |
| DOT | EXIT | # DOT U WITH LOS TO GET TRUNNION COMMAND. |
|     | 32D  |                                           |

# AT THIS POINT WE HAVE A ROTATION VECTOR IN DISH AXES LYING IN THE TS PLANE. CONVERT THIS TO A  
# COMMANDED RATE AND ENABLE THE TRACKER IF WE ARE WITHIN .5 DEGREES OF THE TARGET.

CS MPAC # DOT WAS NEGATIVE OF DESREG ANGLE.

EXTEND

MP RDESGAIN # SCALING ON INPUT ANGLE WAS 4 RADIAN.

TS TANG # TRUNNION COMMAND.

CS RADMODES # A RELAY IN THE RR REVERSES POLARITY OF

MASK BIT12 # THE SHAFT COMMANDS IN MODE 2 SO THAT A

EXTEND # POSITIVE TORQUE APPLIED TO THE SHAFT

BZF +3 # GYRO CAUSES A POSITIVE CHANGE IN THE

CA TANG +1 # SHAFT ANGLE. COMPENSATE FOR THIS SWITCH

TCF +2 # BY CHANGING THE POLARITY OF OUR COMMAND.

+3 CS TANG +1

EXTEND

MP RDESGAIN # SCALING ON INPUT ANGLE WAS 4 RADIAN.

TS TANG +1 # SHAFT COMMAND FOR RR OUT

TC INTPRET

DLOAD DMP

2 # COS(S).

4 # COS(T).

SL1 PDDL # Z COMPONENT OF URR.

DCOMP PDDL # Y COMPONENT = -SIN(T)

0 # SIN(S).

DMP SL1

4 # COS(T).

VDEF BON # FORM URR IN NB AXES.

RRNBSW # BYPASS NBSM CONVERSION IN VERB 41

+3

CALL

\*NBSM\* # GET URR IN SM AXES.

DOT EXIT

RRTARGET # GET COSIN OF ANGLE BETWEEN RR AND LOS

EXTEND

DCS COS1/2DG

DAS MPAC # DIFFERENCE OF COSINES, SCALED B-2.

CCS MPAC

CA ZERO # IF COS ERROR BIGGER, ERROR IS SMALLER

TCF +2

CA ONE

TS MPAC +1 # ZERO IF RR IS POINTED OK, ONE IF NOT.

# SEE IF TRACKER SHOULD BE ENABLED OR DISABLED.

|  |     |          |                                           |
|--|-----|----------|-------------------------------------------|
|  | CCS | RADMODES | # IF CONTINUOUS DESIGNATE WANTED, PUT OUT |
|  | TCF | SIGNLCHK | # COMMANDS WITHOUT CHECKING MAGNITUDE OF  |
|  | TCF | SIGNLCHK | # ERROR SIGNALS                           |

|          |     |          |                                         |
|----------|-----|----------|-----------------------------------------|
| SIGNLCHK | TCF | DORROUT  |                                         |
|          | CCS | MPAC +1  | # SEE IF BOTH AXES WERE WITHIN .5 DEGS. |
|          | TCF | DGOODCHK |                                         |

|  |      |          |                                           |
|--|------|----------|-------------------------------------------|
|  | CS   | STATE    | # IF WITHIN LIMITS AND NO LOCK-ON WANTED, |
|  | MASK | LOKONBIT | # PROBLEM IS FINISHED.                    |
|  | CCS  | A        |                                           |

|  |     |          |  |
|--|-----|----------|--|
|  | TCF | RRDESDUN |  |
|--|-----|----------|--|

|  |     |       |                      |
|--|-----|-------|----------------------|
|  | CAF | BIT14 | # ENABLE THE TRACKER |
|--|-----|-------|----------------------|

|  |        |        |  |
|--|--------|--------|--|
|  | EXTEND |        |  |
|  | WOR    | CHAN12 |  |

|          |     |      |                                 |
|----------|-----|------|---------------------------------|
| DGOODCHK | CAF | BIT4 | # SEE IF DATA GOOD RECEIVED YET |
|----------|-----|------|---------------------------------|

|  |        |        |  |
|--|--------|--------|--|
|  | EXTEND |        |  |
|  | RAND   | CHAN33 |  |

|  |     |         |  |
|--|-----|---------|--|
|  | CCS | A       |  |
|  | TCF | DORROUT |  |

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| RRDESDUN | CS     | BIT10    | # WHEN PROBLEM DONE, REMOVE BIT 10 SO NEXT |
|          | MASK   | RADMODES | # WAITLIST TASK WE WILL GO TO RGOODEND.    |
|          | INHINT |          |                                            |

|  |    |          |  |
|--|----|----------|--|
|  | TS | RADMODES |  |
|--|----|----------|--|

|  |    |          |                              |
|--|----|----------|------------------------------|
|  | TC | DOWNFLAG | # RESET LOSCMFLG TO PREENT A |
|--|----|----------|------------------------------|

|  |        |          |                                        |
|--|--------|----------|----------------------------------------|
|  | ADRES  | LOSCMFLG | # RECOMPUTATION OF LOS AFTER DATA GOOD |
|  | CS     | BIT2     | # TURN OFF ENABLE RR ERROR COUNTER     |
|  | EXTEND |          |                                        |

|  |      |          |                       |
|--|------|----------|-----------------------|
|  | WAND | CHAN12   |                       |
|  | TCF  | ENDOFJOB | # WITH ECTR DISABLED. |

|         |        |          |                                           |
|---------|--------|----------|-------------------------------------------|
| DORROUT | CA     | FLAGWRD2 | # IF BOTH LOSCMFLAG AND SEARCH FLAG ARE   |
|         | MASK   | BIT12,14 | # ZERO, BYPASS VELOCITY ADJUSTMENT TO LOS |
|         | EXTEND |          |                                           |

|  |       |         |                                  |
|--|-------|---------|----------------------------------|
|  | BZF   | NOTP20  |                                  |
|  | TC    | INTPRET |                                  |
|  | VLOAD | VXSC    | # MULTIPLY UNIT LOS BY MAGNITUDE |

|  |      |          |  |
|--|------|----------|--|
|  |      | RRTARGET |  |
|  |      | MLOSV    |  |
|  | VSL1 | PUSH     |  |

|  |       |        |                           |
|--|-------|--------|---------------------------|
|  | VLOAD | VXSC   | # ADD .5 SEC. OF VELOCITY |
|  |       | LOSVEL | # TO LOS VECTOR           |
|  |       | MCTOMS |                           |

|  |      |     |  |
|--|------|-----|--|
|  | VSR1 | VAD |  |
|--|------|-----|--|

|  |       |          |                                       |
|--|-------|----------|---------------------------------------|
|  | UNIT  |          |                                       |
|  | STODL | RRTARGET | # STORE VELOCITY-CORRECTED LOS (UNIT) |



| 14 | 12 | THE |
|----|----|-----|
| 1  |    |     |
| 2  |    |     |
| 3  |    |     |
| 4  |    |     |
| 5  |    |     |
| 6  |    |     |
| 7  |    |     |
| 8  |    |     |
| 9  |    |     |
| 10 |    |     |
| 11 |    |     |
| 12 |    |     |
| 13 |    |     |
| 14 |    |     |
| 15 |    |     |
| 16 |    |     |
| 17 |    |     |
| 18 |    |     |
| 19 |    |     |
| 20 |    |     |
| 21 |    |     |
| 22 |    |     |
| 23 |    |     |
| 24 |    |     |
| 25 |    |     |
| 26 |    |     |
| 27 |    |     |
| 28 |    |     |
| 29 |    |     |
| 30 |    |     |
| 31 |    |     |
| 32 |    |     |
| 33 |    |     |
| 34 |    |     |
| 35 |    |     |
| 36 |    |     |
| 37 |    |     |
| 38 |    |     |
| 39 |    |     |
| 40 |    |     |
| 41 |    |     |
| 42 |    |     |
| 43 |    |     |
| 44 |    |     |
| 45 |    |     |
| 46 |    |     |
| 47 |    |     |
| 48 |    |     |
| 49 |    |     |
| 50 |    |     |
| 51 |    |     |
| 52 |    |     |
| 53 |    |     |
| 54 |    |     |
| 55 |    |     |
| 56 |    |     |
| 57 |    |     |
| 58 |    |     |
| 59 |    |     |
| 60 |    |     |
| 61 |    |     |
| 62 |    |     |
| 63 |    |     |
| 64 |    |     |
| 65 |    |     |
| 66 |    |     |
| 67 |    |     |
| 68 |    |     |
| 69 |    |     |
| 70 |    |     |
| 71 |    |     |
| 72 |    |     |
| 73 |    |     |
| 74 |    |     |
| 75 |    |     |
| 76 |    |     |
| 77 |    |     |
| 78 |    |     |
| 79 |    |     |
| 80 |    |     |



|    |                                                                         |       |             |                                      |    |
|----|-------------------------------------------------------------------------|-------|-------------|--------------------------------------|----|
| 1  | # RADAR READ INITIALIZATION                                             |       |             |                                      | 1  |
| 2  | #                                                                       |       |             |                                      | 2  |
| 3  | # RADAR DATA READ BY A BANKCALL FOR THE APPROPRIATE LEAD-IN BELOW.      |       |             |                                      | 3  |
| 4  |                                                                         |       |             |                                      | 4  |
| 5  |                                                                         |       |             |                                      | 5  |
| 6  | LRALT                                                                   | TC    | INITREAD -1 | # ONE SAMPLE PER READING.            | 6  |
| 7  | ALLREAD                                                                 | OCT   | 17          |                                      | 7  |
| 8  |                                                                         |       |             |                                      | 8  |
| 9  | LRVELZ                                                                  | TC    | INITREAD    |                                      | 9  |
| 10 |                                                                         | OCT   | 16          |                                      | 10 |
| 11 |                                                                         |       |             |                                      | 11 |
| 12 | LRVELY                                                                  | TC    | INITREAD    |                                      | 12 |
| 13 |                                                                         | OCT   | 15          |                                      | 13 |
| 14 |                                                                         |       |             |                                      | 14 |
| 15 | LRVELX                                                                  | TC    | INITREAD    |                                      | 15 |
| 16 |                                                                         | OCT   | 14          |                                      | 16 |
| 17 |                                                                         |       |             |                                      | 17 |
| 18 | RRRDOT                                                                  | TC    | INITREAD -1 |                                      | 18 |
| 19 |                                                                         | OCT   | 12          |                                      | 19 |
| 20 |                                                                         |       |             |                                      | 20 |
| 21 | RRRRANGE                                                                | TC    | INITREAD -1 |                                      | 21 |
| 22 |                                                                         | OCT   | 11          |                                      | 22 |
| 23 |                                                                         |       |             |                                      | 23 |
| 24 | # LRVEL IS THE ENTRY TO THE LR VELOCITY READ ROUTINE WHEN 5 SAMPLES ARE |       |             |                                      | 24 |
| 25 | # WANTED. ENTER WITH C(A)= 0,2,4 FOR LRVELZ,LRVELY,LRVELX RESP.         |       |             |                                      | 25 |
| 26 |                                                                         |       |             |                                      | 26 |
| 27 | LRVEL                                                                   | TS    | TIMEHOLD    | # STORE VBEAM INDEX HERE MOMENTARILY | 27 |
| 28 |                                                                         | CAF   | FIVE        | # SPECIFY FIVE SAMPLES               | 28 |
| 29 |                                                                         | INDEX | TIMEHOLD    |                                      | 29 |
| 30 |                                                                         | TCF   | LRVELZ      |                                      | 30 |
| 31 |                                                                         |       |             |                                      | 31 |
| 32 |                                                                         |       |             |                                      | 32 |
| 33 |                                                                         |       |             |                                      | 33 |
| 34 |                                                                         |       |             |                                      | 34 |
| 35 |                                                                         |       |             |                                      | 35 |
| 36 |                                                                         |       |             |                                      | 36 |
| 37 |                                                                         |       |             |                                      | 37 |
| 38 |                                                                         |       |             |                                      | 38 |
| 39 |                                                                         |       |             |                                      | 39 |
| 40 |                                                                         |       |             |                                      | 40 |
| 41 |                                                                         |       |             |                                      | 41 |
| 42 |                                                                         |       |             |                                      | 42 |
| 43 |                                                                         |       |             |                                      | 43 |
| 44 |                                                                         |       |             |                                      | 44 |
| 45 |                                                                         |       |             |                                      | 45 |
| 46 |                                                                         |       |             |                                      | 46 |
| 47 |                                                                         |       |             |                                      | 47 |
| 48 |                                                                         |       |             |                                      | 48 |
| 49 |                                                                         |       |             |                                      | 49 |
| 50 |                                                                         |       |             |                                      | 50 |
| 51 |                                                                         |       |             |                                      | 51 |
| 52 |                                                                         |       |             |                                      | 52 |
| 53 |                                                                         |       |             |                                      | 53 |
| 54 |                                                                         |       |             |                                      | 54 |
| 55 |                                                                         |       |             |                                      | 55 |
| 56 |                                                                         |       |             |                                      | 56 |
| 57 |                                                                         |       |             |                                      | 57 |
| 58 |                                                                         |       |             |                                      | 58 |
| 59 |                                                                         |       |             |                                      | 59 |
| 60 |                                                                         |       |             |                                      | 60 |



|    |                                                               |        |          |                                            |    |
|----|---------------------------------------------------------------|--------|----------|--------------------------------------------|----|
| 1  |                                                               |        |          |                                            | 1  |
| 2  | -1                                                            | CAF    | ONE      | # ENTRY TO TAKE ONLY 1 SAMPLE              | 2  |
| 3  | INITREAD                                                      | INHINT |          |                                            | 3  |
| 4  |                                                               |        |          |                                            | 4  |
| 5  |                                                               | TS     | TIMEHOLD | # GET DT OF MIDPOINT OF NOMINAL SAMPLING   | 5  |
| 6  |                                                               | EXTEND |          | # INTERVAL (ASSUMES NO BAD SAMPLES WILL BE | 6  |
| 7  |                                                               | MP     | BIT3     | # ENCOUNTERED).                            | 7  |
| 8  |                                                               | DXCH   | TIMEHOLD |                                            | 8  |
| 9  |                                                               |        |          |                                            | 9  |
| 10 |                                                               | CCS    | A        |                                            | 10 |
| 11 |                                                               | TS     | NSAMP    |                                            | 11 |
| 12 |                                                               | AD     | ONE      |                                            | 12 |
| 13 | # INSERT FOLLOWING INSTRUCTION TO GET 2N TRIES FOR N SAMPLES. |        |          |                                            | 13 |
| 14 | #                                                             | DOUBLE |          |                                            | 14 |
| 15 |                                                               | TS     | SAMPLIM  |                                            | 15 |
| 16 |                                                               |        |          |                                            | 16 |
| 17 |                                                               | CAF    | DGBITS   | # READ CURRENT VALUE OF DATA GOOD BITS.    | 17 |
| 18 |                                                               | EXTEND |          |                                            | 18 |
| 19 |                                                               | RAND   | CHAN33   |                                            | 19 |
| 20 |                                                               | TS     | OLDATAGD |                                            | 20 |
| 21 |                                                               |        |          |                                            | 21 |
| 22 |                                                               | CS     | ALLREAD  |                                            | 22 |
| 23 |                                                               | EXTEND |          |                                            | 23 |
| 24 |                                                               | WAND   | CHAN13   | # REMOVE ALL RADAR BITS                    | 24 |
| 25 |                                                               |        |          |                                            | 25 |
| 26 |                                                               | INDEX  | Q        |                                            | 26 |
| 27 |                                                               | CAF    | 0        |                                            | 27 |
| 28 |                                                               | EXTEND |          |                                            | 28 |
| 29 |                                                               | WOR    | CHAN13   | # SET NEW RADAR BITS                       | 29 |
| 30 |                                                               |        |          |                                            | 30 |
| 31 |                                                               | EXTEND |          |                                            | 31 |
| 32 |                                                               | DCA    | TIME2    |                                            | 32 |
| 33 |                                                               | DAS    | TIMEHOLD | # TIME OF NOMINAL MIDPOINT                 | 33 |
| 34 |                                                               |        |          |                                            | 34 |
| 35 |                                                               | CAF    | ZERO     |                                            | 35 |
| 36 |                                                               | TS     | L        |                                            | 36 |
| 37 |                                                               | DXCH   | SAMPLSUM |                                            | 37 |
| 38 |                                                               | TCF    | ROADBACK |                                            | 38 |
| 39 |                                                               |        |          |                                            | 39 |
| 40 | DGBITS                                                        | OCT    | 230      |                                            | 40 |
| 41 |                                                               |        |          |                                            | 41 |
| 42 |                                                               |        |          |                                            | 42 |
| 43 |                                                               |        |          |                                            | 43 |
| 44 |                                                               |        |          |                                            | 44 |
| 45 |                                                               |        |          |                                            | 45 |
| 46 |                                                               |        |          |                                            | 46 |
| 47 |                                                               |        |          |                                            | 47 |
| 48 |                                                               |        |          |                                            | 48 |
| 49 |                                                               |        |          |                                            | 49 |
| 50 |                                                               |        |          |                                            | 50 |
| 51 |                                                               |        |          |                                            | 51 |
| 52 |                                                               |        |          |                                            | 52 |
| 53 |                                                               |        |          |                                            | 53 |
| 54 |                                                               |        |          |                                            | 54 |
| 55 |                                                               |        |          |                                            | 55 |
| 56 |                                                               |        |          |                                            | 56 |
| 57 |                                                               |        |          |                                            | 57 |
| 58 |                                                               |        |          |                                            | 58 |
| 59 |                                                               |        |          |                                            | 59 |
| 60 |                                                               |        |          |                                            | 60 |

# RADAR RUPT READER

#

# THIS ROUTINE STARTS FROM A RADARUPT. IT READS THE DATA &amp; LOTS MORE.

SETLOC RADARUPT  
BANK

COUNT\* \$\$/RRUPT

RADAREAD EXTEND # MUST SAVE SBANK BECAUSE OF RUPT EXITS  
ROR SUPERBNK # VIA TASKOVER (BADEND OR GOODEND).  
TS BANKRUPTEXTEND  
QXCH QRUPT

CAF SEVEN

EXTEND  
RAND CHAN13

TS DNINDEX

EXTEND # IF RADAR SELECT BITS ZERO, DO NOT STORE  
BZF TRYCOUNT # DATA FOR DOWNLIST (ERASABLE PROBLEMS)CA RNRAD  
INDEX DNINDEX  
TS DNRRANGE -1TRYCOUNT CCS SAMPLIM  
TCF PLENTY  
TCF NOMORETC ALARM  
OCT 520  
TC RESUMENOMORE CA FLGWRD11 # IS LRBYPASS SET?  
MASK LRBYBITEXTEND  
BZF BADRAD # NO. R12 IS ON -- BYPASS 521 ALARM.CS FLAGWRD3 # CHECK R04FLAG.  
MASK R04FLBIT # IF 1, R04 IS RUNNING. DO NOT ALARMEXTEND  
BZF BADRAD

TC ALARM # P20 WANTS THE ALARM.

BADRAD OCT 521  
CS ONE  
TS SAMPLIMPLENTY TC RDBADEND -2  
TS SAMPLIM  
CAF BIT3EXTEND  
RAND CHAN13 # TO FIND OUT WHICH RADAR  
EXTEND

|                   |                                |                                           |                                                                                                                        |
|-------------------|--------------------------------|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
|                   | BZF                            | RENDRAD                                   |                                                                                                                        |
| LRPOSCHK          | TC<br>CA<br>EXTEND             | R77CHECK<br>RADMODES                      | # R77 QUILTS HERE.<br># SEE IF LR IN DESIRED POSITION                                                                  |
|                   | RXOR<br>MASK<br>EXTEND         | CHAN33<br>BIT6                            |                                                                                                                        |
|                   | BZF                            | VELCHK                                    |                                                                                                                        |
|                   | TC<br>OCT<br>TC                | ALARM<br>522<br>BADRAD                    |                                                                                                                        |
| VELCHK            | CAF<br>EXTEND<br>RXOR          | BIN3<br>CHAN13                            | # = 00003 OCT<br># RESET ACTIVITY BIT                                                                                  |
|                   | MASK<br>EXTEND<br>BZF          | BIN3<br>LRHEIGHT                          | # TAKE A LR RANGE READING                                                                                              |
|                   | CAF<br>MASK<br>AD<br>TS<br>CAE | POSMAX<br>RNRAD<br>LVELBIAS<br>L<br>RNRAD |                                                                                                                        |
|                   | DOUBLE<br>MASK<br>DXCH         | BIT1<br>ITEMP3                            |                                                                                                                        |
|                   | CAF<br>TC                      | BIT8<br>DGCHECK                           | # DATA GOOD ISN'T CHECKED UNTIL AFTER READ-<br># ING DATA SO SOME RADAR TESTS WILL WORK<br># INDEPENDENT OF DATA GOOD. |
| GOODRAD           | CCS<br>TC<br>CS<br>TS          | NSAMP<br>NOEND<br>ONE<br>SAMPLIM          |                                                                                                                        |
|                   | CS<br>MASK<br>TS               | ITEMP1<br>RADMODES<br>RADMODES            | # WHEN ENOUGH GOOD DATA HAS BEEN GATHERED,<br># RESET DATA FAIL FLAGS FOR SETTRKF.                                     |
|                   | TC<br>TC                       | RADLITES<br>RGOODEND -2                   | # LAMPS MAY GO OFF IF DATA JUST GOOD.                                                                                  |
| NOEND<br>RESAMPLE | TS<br>CCS<br>TCF               | NSAMP<br>SAMPLIM<br>+2                    | # SEE IF ANY MORE TRIES SHOULD BE MADE.                                                                                |
|                   | TCF<br>CAF<br>EXTEND           | DATAFAIL<br>BIT4                          | # N SAMPLES NOT AVAILABLE.<br># RESET ACTIVITY BIT.                                                                    |

|    |           |        |             |                                          |
|----|-----------|--------|-------------|------------------------------------------|
| 1  | # FLS FLS |        |             | PAGE 331                                 |
| 2  |           | WOR    | CHAN13      | # RESET ACTIVITY BIT                     |
| 3  |           | TC     | RESUME      |                                          |
| 4  |           |        |             |                                          |
| 5  |           |        |             |                                          |
| 6  | LRHEIGHT  | CAF    | BIT5        |                                          |
| 7  |           | TS     | ITEMP1      | # (POSITION OF DATA GOOD BIT IN CHAN 33) |
| 8  |           |        |             |                                          |
| 9  |           | CAF    | BIT9        |                                          |
| 10 |           | TC     | SCALECHK -1 |                                          |
| 11 |           |        |             |                                          |
| 12 | RENDRAD   | CAF    | REPOSBIT    | # MAKE SURE ANTENNA HAS NOT GONE OUT OF  |
| 13 |           | MASK   | RADMODES    | # LIMITS.                                |
| 14 |           | CCS    | A           |                                          |
| 15 |           | TCF    | BADRAD      |                                          |
| 16 |           |        |             |                                          |
| 17 |           | CS     | RADMODES    | # BE SURE RR CDU HASN'T FAILED.          |
| 18 |           | MASK   | RCDUFBIT    |                                          |
| 19 |           | CCS    | A           |                                          |
| 20 |           | TCF    | BADRAD      |                                          |
| 21 |           |        |             |                                          |
| 22 |           | CAF    | BIT4        | # SEE IF DATA HAS BEEN GOOD.             |
| 23 |           | TS     | ITEMP1      | # (POSITION OF DATA GOOD BIT IN CHAN 33) |
| 24 |           |        |             |                                          |
| 25 |           | CAF    | BIT1        | # SEE IF RR RDOT.                        |
| 26 |           | EXTEND |             |                                          |
| 27 |           | RAND   | CHAN13      |                                          |
| 28 |           | TS     | Q           | # FOR LATER TESTING.                     |
| 29 |           | CCS    | A           |                                          |
| 30 |           | TCF    | +2          |                                          |
| 31 |           | TCF    | RADIN       | # NO SCALE CHECK FOR RR RDOT.            |
| 32 |           | CAF    | BIT3        |                                          |
| 33 |           | TS     | L           |                                          |
| 34 |           |        |             |                                          |
| 35 | SCALECHK  | EXTEND |             |                                          |
| 36 |           | RAND   | CHAN33      | # SCALE STATUS NOW                       |
| 37 |           | XCH    | L           |                                          |
| 38 |           | MASK   | RADMODES    | # SCALE STATUS BEFORE                    |
| 39 |           | EXTEND |             |                                          |
| 40 |           | RXOR   | LCHAN       | # SEE IF THEY DIFFER                     |
| 41 |           | CCS    | A           |                                          |
| 42 |           | TC     | SCALCHNG    | # THEY DIFFER.                           |
| 43 |           |        |             |                                          |
| 44 | RADIN     | CAF    | POSMAX      |                                          |
| 45 |           | MASK   | RNRAD       |                                          |
| 46 |           | TS     | ITEMP4      |                                          |
| 47 |           |        |             |                                          |
| 48 |           | CAE    | RNRAD       |                                          |
| 49 |           | DOUBLE |             |                                          |
| 50 |           | MASK   | BIT1        |                                          |
| 51 |           | TS     | ITEMP3      |                                          |
| 52 |           |        |             |                                          |
| 53 |           |        |             |                                          |
| 54 |           |        |             |                                          |
| 55 |           |        |             |                                          |
| 56 |           |        |             |                                          |
| 57 |           |        |             |                                          |
| 58 |           |        |             |                                          |
| 59 |           |        |             |                                          |
| 60 |           |        |             |                                          |

|    |                                                                  |        |             |                                          |
|----|------------------------------------------------------------------|--------|-------------|------------------------------------------|
| 1  | # PLS PLS                                                        |        |             |                                          |
| 2  |                                                                  | CCS    | Q           | # SEE IF RR RDOT.                        |
| 3  |                                                                  | TCF    | SCALADJ     | # NO, BUT SCALE CHANGING MAY BE NEEDED.  |
| 4  |                                                                  |        |             |                                          |
| 5  |                                                                  | EXTEND |             | # IF RR RANGE RATE, THROW OUT BIAS       |
| 6  |                                                                  | DCS    | RDOTBIAS    |                                          |
| 7  | DASAMPL                                                          | DAS    | ITEMP3      |                                          |
| 8  | DGCHECK2                                                         | CA     | ITEMP1      | # SEE THAT DATA HAS BEEN GOOD BEFORE AND |
| 9  |                                                                  | TC     | DGCHECK +1  | # AFTER TAKING SAMPLE.                   |
| 10 |                                                                  | TC     | GOODRAD     |                                          |
| 11 |                                                                  |        |             |                                          |
| 12 | SCALCHNG                                                         | LXCH   | RADMODES    |                                          |
| 13 |                                                                  | AD     | BIT1        |                                          |
| 14 |                                                                  | EXTEND |             |                                          |
| 15 |                                                                  | RXOR   | LCHAN       |                                          |
| 16 |                                                                  | TS     | RADMODES    |                                          |
| 17 |                                                                  | CAF    | DGBITS      | # UPDATE LAST VALUE OF DATA GOOD BITS.   |
| 18 |                                                                  | EXTEND |             |                                          |
| 19 |                                                                  | RAND   | CHAN33      |                                          |
| 20 |                                                                  | TS     | OLDATAGD    |                                          |
| 21 |                                                                  | TC     | UPFLAG      | # SET RNGSCFLG                           |
| 22 |                                                                  | ADRES  | RNGSCFLG    | # FOR LRS24.1                            |
| 23 |                                                                  | TCF    | BADRAD      |                                          |
| 24 |                                                                  |        |             |                                          |
| 25 | # R77 MUST IGNORE DATA FAILS SO AS NOT TO DISTURB THE ASTRONAUT. |        |             |                                          |
| 26 |                                                                  |        |             |                                          |
| 27 | R77CHECK                                                         | CS     | FLAGWRD5    |                                          |
| 28 |                                                                  | MASK   | R77FLBIT    |                                          |
| 29 |                                                                  | CCS    | A           |                                          |
| 30 |                                                                  | TC     | Q           | # NOT R77                                |
| 31 |                                                                  | CS     | BITS5,8     | # UPDATE LR DATA GOOD BITS IN RADMODES   |
| 32 |                                                                  | MASK   | RADMODES    |                                          |
| 33 |                                                                  | TS     | L           |                                          |
| 34 |                                                                  | CA     | BITS5,8     |                                          |
| 35 |                                                                  | EXTEND |             |                                          |
| 36 |                                                                  | RAND   | CHAN33      |                                          |
| 37 |                                                                  | AD     | L           |                                          |
| 38 |                                                                  | TS     | RADMODES    |                                          |
| 39 |                                                                  | TC     | RGOODEND -2 |                                          |
| 40 | BITS5,8                                                          | OCT    | 220         |                                          |

# THE FOLLOWING ROUTINE INCORPORATES RR RANGE AND LR ALT SCALE INFORMATION AND LEAVES DATA AT LO SCALE.

|          |        |          |                                     |
|----------|--------|----------|-------------------------------------|
| SCALADJ  | CCS    | L        | # L HAS SCALE INBIT FOR THIS RADAR. |
|          | TCF    | +2       | # ON HIGH SCALE.                    |
|          | TCF    | DGCHECK2 |                                     |
|          | CA     | DNINDEX  |                                     |
|          | MASK   | BIT3     |                                     |
|          | CCS    | A        |                                     |
|          | TCF    | LRSCK    |                                     |
|          | DXCH   | ITEMP3   |                                     |
|          | DDOUBL |          |                                     |
|          | DDOUBL |          |                                     |
|          | DDOUBL |          |                                     |
|          | DXCH   | ITEMP3   |                                     |
|          | TCF    | DGCHECK2 |                                     |
| LRSCK    | CCS    | ITEMP3   |                                     |
|          | TCF    | +11      |                                     |
|          | CS     | ITEMP4   |                                     |
|          | AD     | HISCALIM |                                     |
|          | EXTEND |          |                                     |
|          | BZMF   | +5       |                                     |
|          | CS     | FLGWRD11 |                                     |
|          | MASK   | SCABBIT  |                                     |
|          | ADS    | FLGWRD11 |                                     |
|          | TCF    | +4       |                                     |
|          | CS     | SCABBIT  |                                     |
|          | MASK   | FLGWRD11 |                                     |
|          | TS     | FLGWRD11 |                                     |
|          | EXTEND |          |                                     |
|          | DCA    | ITEMP3   |                                     |
|          | DDOUBL |          |                                     |
|          | DDOUBL |          |                                     |
|          | TCF    | DASAMPL  |                                     |
| HISCALIM | DEC    | 460      | # 2481.7 FT *****                   |



|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  | DGCHECK  | TS     | ITEMP1   | # UPDATE DATA GOOD BIT IN OLDTAGD AND      | 2  |
| 3  |          | EXTEND |          | # MAKE SURE IT WAS ON BEFORE AND AFTER THE | 3  |
| 4  |          | RAND   | CHAN33   | # SAMPLE WAS TAKEN BEFORE RETURNING. IF    | 4  |
| 5  |          | TS     | L        | # NOT, GOES TO RESAMPLE TO TRY AGAIN. IF   | 5  |
| 6  |          | CS     | ITEMP1   | # MAX NUMBER OF TRIES HAS BEEN REACHED,    | 6  |
| 7  |          | MASK   | OLDTAGD  | # THE BIT CORRESPONDING TO THE DATA GOOD   | 7  |
| 8  |          | AD     | L        | # WHICH FAILED TO APPEAR IS IN ITEMP1 AND  | 8  |
| 9  |          | XCH    | OLDTAGD  | # CAN BE USED TO SET RADMODES WHICH VIA    | 9  |
| 10 |          | MASK   | ITEMP1   | # SETTRKF SETS THE TRACKER FAIL LAMP.      | 10 |
| 11 |          | AD     | L        |                                            | 11 |
| 12 |          | CCS    | A        | # SHOULD BOTH BE ZERO.                     | 12 |
| 13 |          | TC     | RESAMPLE |                                            | 13 |
| 14 |          | DXCH   | ITEMP3   | # IF DATA GOOD BEFORE AND AFTER, ADD TO    | 14 |
| 15 |          | DAS    | SAMPLSUM | # ACCUMULATION.                            | 15 |
| 16 |          | TC     | Q        |                                            | 16 |
| 17 |          |        |          |                                            | 17 |
| 18 | DATAFAIL | CS     | ITEMP1   | # IN THE ABOVE CASE, SET RADMODES BIT      | 18 |
| 19 |          | MASK   | RADMODES | # SHOWING SOME RADAR DATA FAILED.          | 19 |
| 20 |          | AD     | ITEMP1   |                                            | 20 |
| 21 |          | TS     | RADMODES |                                            | 21 |
| 22 |          |        |          |                                            | 22 |
| 23 |          | DXCH   | ITEMP3   | # IF WE HAVE BEEN UNABLE TO GATHER N       | 23 |
| 24 |          | DXCH   | SAMPLSUM | # SAMPLES, USE LAST ONE ONLY.              | 24 |
| 25 |          | TC     | RADLITES |                                            | 25 |
| 26 |          | TCF    | NOMORE   |                                            | 26 |
| 27 |          |        |          |                                            | 27 |
| 28 |          |        |          |                                            | 28 |
| 29 |          |        |          |                                            | 29 |
| 30 |          |        |          |                                            | 30 |
| 31 |          |        |          |                                            | 31 |
| 32 |          |        |          |                                            | 32 |
| 33 |          |        |          |                                            | 33 |
| 34 |          |        |          |                                            | 34 |
| 35 |          |        |          |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 |          |        |          |                                            | 37 |
| 38 |          |        |          |                                            | 38 |
| 39 |          |        |          |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 |          |        |          |                                            | 41 |
| 42 |          |        |          |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          |        |          |                                            | 44 |
| 45 |          |        |          |                                            | 45 |
| 46 |          |        |          |                                            | 46 |
| 47 |          |        |          |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          |        |          |                                            | 49 |
| 50 |          |        |          |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

# THIS ROUTINE CHANGES THE LR POSITION, AND CHECKS THAT IT GOT THERE.

SETLOC P20S1  
BANK

LRPOS2

COUNT\* \$\$/RSUB  
INHINT

CS RADMODES  
MASK LRPOSBIT # SHOW DESIRED LR POSITION IS 2  
ADS RADMODES

CAF BIT7  
EXTEND

RAND CHAN33 # SEE IF ALREADY THERE.  
EXTEND  
BZF RADNOOP

CAF BIT13  
EXTEND

WOR CHAN12 # COMMAND TO POSITION 2  
CAF 6SECS # START SCANNING FOR INBIT AFTER 7 SECS.  
TC WAITLIST

EBANK= LOSCOUNT  
2CADR LRPOSCAN

TC ROADBACK

LRPOSNXT

TS SAMPLIM  
TC FIXDELAY # SCAN ONCE PER SECOND 15 TIMES MAX AFTER  
DEC 100 # INITIAL DELAY OF 7 SECONDS.

CAF BIT7 # SEE IF LR POS2 IS ON  
EXTEND  
RAND CHAN33

EXTEND  
BZF LASTLRDT # IF THERE, WAIT FINAL SECOND FOR BOUNCE.

CCS SAMPLIM # SEE IF MAX TIME UP.  
TCF LRPOSNXT

CS BIT13 # IF TIME UP, DISABLE COMMAND AND ALARM.  
EXTEND  
WAND CHAN12  
TCF RDBADEND

RADNOOP

CAF ONE # NO FURTHER ACTION REQUESTED.  
TC WAITLIST  
EBANK= LOSCOUNT  
2CADR RGOODEND



|    |          |        |          |                                           |    |
|----|----------|--------|----------|-------------------------------------------|----|
| 1  |          |        |          |                                           | 1  |
| 2  |          | TC     | ROADBACK |                                           | 2  |
| 3  |          |        |          |                                           | 3  |
| 4  | LASTLRDT | CA     | 2SECS    | # WAIT TWO SECONDS AFTER RECEIPT OF INBIT | 4  |
| 5  |          | TC     | VARDELAY | # TO WAIT FOR ANTENNA BOUNCE TO DIE OUT.  | 5  |
| 6  |          |        |          |                                           | 6  |
| 7  |          | CS     | BIT13    | # REMOVE COMMAND                          | 7  |
| 8  |          | EXTEND |          |                                           | 8  |
| 9  |          | WAND   | CHAN12   |                                           | 9  |
| 10 |          | TCF    | RGOODEND |                                           | 10 |
| 11 |          |        |          |                                           | 11 |
| 12 | LRPOSCAN | CAF    | FOURTEEN | # SET UP FOR 15 SAMPLES.                  | 12 |
| 13 |          | TCF    | LRPOSNXT |                                           | 13 |
| 14 | 6SECS    | DEC    | 600      |                                           | 14 |
| 15 |          |        |          |                                           | 15 |
| 16 |          |        |          |                                           | 16 |
| 17 |          |        |          |                                           | 17 |
| 18 |          |        |          |                                           | 18 |
| 19 |          |        |          |                                           | 19 |
| 20 |          |        |          |                                           | 20 |
| 21 |          |        |          |                                           | 21 |
| 22 |          |        |          |                                           | 22 |
| 23 |          |        |          |                                           | 23 |
| 24 |          |        |          |                                           | 24 |
| 25 |          |        |          |                                           | 25 |
| 26 |          |        |          |                                           | 26 |
| 27 |          |        |          |                                           | 27 |
| 28 |          |        |          |                                           | 28 |
| 29 |          |        |          |                                           | 29 |
| 30 |          |        |          |                                           | 30 |
| 31 |          |        |          |                                           | 31 |
| 32 |          |        |          |                                           | 32 |
| 33 |          |        |          |                                           | 33 |
| 34 |          |        |          |                                           | 34 |
| 35 |          |        |          |                                           | 35 |
| 36 |          |        |          |                                           | 36 |
| 37 |          |        |          |                                           | 37 |
| 38 |          |        |          |                                           | 38 |
| 39 |          |        |          |                                           | 39 |
| 40 |          |        |          |                                           | 40 |
| 41 |          |        |          |                                           | 41 |
| 42 |          |        |          |                                           | 42 |
| 43 |          |        |          |                                           | 43 |
| 44 |          |        |          |                                           | 44 |
| 45 |          |        |          |                                           | 45 |
| 46 |          |        |          |                                           | 46 |
| 47 |          |        |          |                                           | 47 |
| 48 |          |        |          |                                           | 48 |
| 49 |          |        |          |                                           | 49 |
| 50 |          |        |          |                                           | 50 |
| 51 |          |        |          |                                           | 51 |
| 52 |          |        |          |                                           | 52 |
| 53 |          |        |          |                                           | 53 |
| 54 |          |        |          |                                           | 54 |
| 55 |          |        |          |                                           | 55 |
| 56 |          |        |          |                                           | 56 |
| 57 |          |        |          |                                           | 57 |
| 58 |          |        |          |                                           | 58 |
| 59 |          |        |          |                                           | 59 |
| 60 |          |        |          |                                           | 60 |



|    |                                         |        |          |    |
|----|-----------------------------------------|--------|----------|----|
| 1  | # SEQUENCES TO TERMINATE RR OPERATIONS. |        |          | 1  |
| 2  |                                         |        |          | 2  |
| 3  |                                         |        |          | 3  |
| 4  | ENDRADAR                                | CAF    | RCDUFBIT | 4  |
| 5  |                                         | MASK   | RADMODES | 5  |
| 6  |                                         | CCS    | A        | 6  |
| 7  |                                         |        |          | 7  |
| 8  |                                         | TCF    | RGOODEND | 8  |
| 9  | -2                                      | TCF    | RDBADEND | 9  |
| 10 |                                         | CS     | ZERO     | 10 |
| 11 |                                         | TS     | RUPTAGN  | 11 |
| 12 | RGOODEND                                | CAF    | TWO      | 12 |
| 13 |                                         | TC     | POSTJUMP | 13 |
| 14 |                                         | CADR   | GOODEND  | 14 |
| 15 |                                         |        |          | 15 |
| 16 | -2                                      | CS     | ZERO     | 16 |
| 17 |                                         | TS     | RUPTAGN  | 17 |
| 18 | RDBADEND                                | CAF    | TWO      | 18 |
| 19 |                                         | TC     | POSTJUMP | 19 |
| 20 |                                         | CADR   | BADEND   | 20 |
| 21 |                                         |        |          | 21 |
| 22 | BIN3                                    | EQUALS | THREE    | 22 |
| 23 |                                         |        |          | 23 |
| 24 |                                         |        |          | 24 |
| 25 |                                         |        |          | 25 |
| 26 |                                         |        |          | 26 |
| 27 |                                         |        |          | 27 |
| 28 |                                         |        |          | 28 |
| 29 |                                         |        |          | 29 |
| 30 |                                         |        |          | 30 |
| 31 |                                         |        |          | 31 |
| 32 |                                         |        |          | 32 |
| 33 |                                         |        |          | 33 |
| 34 |                                         |        |          | 34 |
| 35 |                                         |        |          | 35 |
| 36 |                                         |        |          | 36 |
| 37 |                                         |        |          | 37 |
| 38 |                                         |        |          | 38 |
| 39 |                                         |        |          | 39 |
| 40 |                                         |        |          | 40 |
| 41 |                                         |        |          | 41 |
| 42 |                                         |        |          | 42 |
| 43 |                                         |        |          | 43 |
| 44 |                                         |        |          | 44 |
| 45 |                                         |        |          | 45 |
| 46 |                                         |        |          | 46 |
| 47 |                                         |        |          | 47 |
| 48 |                                         |        |          | 48 |
| 49 |                                         |        |          | 49 |
| 50 |                                         |        |          | 50 |
| 51 |                                         |        |          | 51 |
| 52 |                                         |        |          | 52 |
| 53 |                                         |        |          | 53 |
| 54 |                                         |        |          | 54 |
| 55 |                                         |        |          | 55 |
| 56 |                                         |        |          | 56 |
| 57 |                                         |        |          | 57 |
| 58 |                                         |        |          | 58 |
| 59 |                                         |        |          | 59 |
| 60 |                                         |        |          | 60 |

```
PROGRAM NAME: LPS20.1 VECTOR EXTRAPOLATION AND LOS COMPUTATION
MOD. NO. 2 BY J.D. COYNE SDC DATE 12-7-66

#
FUNCTIONAL DESCRIPTION:
1) EXTRAPOLATE THE LEM AND CSM VECTORS IN ACCORDANCE WITH THE TIME REFERRED TO IN CALLER + 1.
2) COMPUTES THE LOS VECTOR TO THE CSM, CONVERTS IT TO STABLE MEMBER COORDINATES AND STORES IT IN RRTARGET.
3) COMPUTES THE MAGNITUDE OF TEH LOS VECTOR AND STORES IT IN MLOSV

CALLING SEQUENCE: CALL
LPS20.1
#
SUBROUTINES CALLED:
LEMPREC, CSMPREC
#
NORMAL EXIT: RETURN TO CALLER + 2.
#
ERROR EXITS: NONE
#
ALARMS: NONE
#
OUTPUT:
LOS VECTOR (HALF UNIT) IN SM COORDINATES STORED IN RRTARGET
MAGNITUDE OF TEH LOS VECTOR (METERS SCALED B-29) STORED IN MSLOV
RRNBSW CLEARED.
#
INITIALIZED ERASABLE
TDEC1 MUST CONTAIN THE TIME FOR EXTRAPOLATION
SEE ORBITAL INTEGRATION ROUTINE
#
DEBRIS:
MPAC DESTROYED BY THE ROUTINE
```

```
BANK 23
SETLOC P20S
BANK
```

|    |          |        |            |                                            |
|----|----------|--------|------------|--------------------------------------------|
| 1  |          |        |            |                                            |
| 2  |          | COUNT* | \$\$/LPS20 |                                            |
| 3  |          |        |            |                                            |
| 4  | LPS20.1  | STQ    | BOFF       |                                            |
| 5  |          |        | LS21X      |                                            |
| 6  |          |        | LOSCMFLG   | # LOSCMFLG = 0 MEANS NOT CALLED BY R21     |
| 7  |          |        | LMINT      | # SO CALL LEMCONIC TO GET LM STATE         |
| 8  |          | BON    |            | # IF IN R21 AND ON LUNAR SURFACE           |
| 9  |          |        | SURFFLAG   | # DON'T CALL LEMCONIC                      |
| 10 |          |        | CSMINT     |                                            |
| 11 | LMINT    | CALL   |            |                                            |
| 12 |          |        | LEMCONIC   | # EXTRAPOLATE LEM                          |
| 13 |          | VLOAD  |            |                                            |
| 14 |          |        | RATT       |                                            |
| 15 |          | STOVL  | LMPOS      | # SAVE LM POSITION B-29                    |
| 16 |          |        | VATT       |                                            |
| 17 |          | STODL  | LMVEL      | # SAVE LM VELOCITY B-7                     |
| 18 |          |        | TAT        |                                            |
| 19 | CSMINT   | STCALL | TDEC1      |                                            |
| 20 |          |        | CSMCONIC   | # EXTRAPOLATE CSM                          |
| 21 |          | VLOAD  | VSU        | # COMPUTE RELATIVE VELOCITY V(CSM) - V(LM) |
| 22 |          |        | VATT       |                                            |
| 23 |          |        | LMVEL      |                                            |
| 24 |          | MXV    | VSL1       |                                            |
| 25 |          |        | REFSMMAT   |                                            |
| 26 |          | EXIT   |            |                                            |
| 27 |          | TC     | KILLTASK   | # KILL THE TASK WHICH CALLS DODES SINCE    |
| 28 |          | CADR   | DESLOOP +2 | # STORING INTO ERASEABLES DODES USES       |
| 29 |          | TC     | INTPRET    |                                            |
| 30 |          | STOVL  | LOSVEL     |                                            |
| 31 |          |        | RATT       |                                            |
| 32 |          | VSU    | BOFF       |                                            |
| 33 |          |        | LMPOS      |                                            |
| 34 |          |        | RNDVZFLG   |                                            |
| 35 |          |        | NOTSHIFT   |                                            |
| 36 |          | BOVB   |            |                                            |
| 37 |          |        | TCDANZIG   |                                            |
| 38 |          | VSL    |            |                                            |
| 39 |          |        | 9D         |                                            |
| 40 | NOTSHIFT | UNIT   | BOVB       | # IF OVERFLOW, RANGE MUST BE GREATER       |
| 41 |          |        | 526ALARM   | # THAN 400 N. M.                           |
| 42 |          | MXV    | VSL1       |                                            |
| 43 |          |        | REFSMMAT   | # CONVERT TO STABLE MEMBER                 |
| 44 |          | STODL  | RRTARGET   |                                            |
| 45 |          |        | 36D        | # SAVE MAGNITUDE OF LOS VECTOR FOR         |
| 46 |          | STORE  | MLOSV      | # VELOCITY CORRECTION IN DESIGNATE         |
| 47 |          | CLRGO  |            |                                            |
| 48 |          |        | RRNBSW     |                                            |
| 49 |          |        | LS21X      |                                            |
| 50 |          |        |            |                                            |
| 51 |          |        |            |                                            |
| 52 |          |        |            |                                            |
| 53 |          |        |            |                                            |
| 54 |          |        |            |                                            |
| 55 |          |        |            |                                            |
| 56 |          |        |            |                                            |
| 57 |          |        |            |                                            |
| 58 |          |        |            |                                            |
| 59 |          |        |            |                                            |
| 60 |          |        |            |                                            |

# P20-P25

# PROGRAM NAME: LPS20.2 400 NM RANGE CHECK  
# MOD. NO. 2 BY J.D. COYNE SDC DATE 12-7-66  
#  
# FUNCTIONAL DESCRIPTION:  
# COMPARES THE MAGNITUDE OF THE LOS VECTOR TO 400 NM.  
#  
# CALLING SEQUENCE: CALL LPS20.2  
#  
# SUBROUTINES CALLED: NONE  
#  
# NORMAL EXIT: RETURN TO CALLER +1, MPAC EQ 0 (RANGE 400NM OR LESS.)  
#  
# ERROR EXITS: RETURN TO CALLER +1, MPAC EQ 1 (RANGE GREATER THAN 400NM)  
#  
# ALARMS: NONE  
#  
# OUTPUT: NONE  
#  
# INITIALIZED ERASEABLE:  
# PDL 36D MUST CONTAIN THE MAGNITUDE OF THE VECTOR  
#  
# DEBRIS:  
# MPAC DESTROYED BY THIS ROUTINE

SETLOC P20S1  
BANK  
COUNT\* \$\$/LPS20

LPS20.2 DLOAD DSU  
MLOSV # MAGNITUDE OF LOS  
FHNM # OVER 400NM

BPL  
TOFAR  
SLOAD RVQ  
ZERO/SP  
TOFAR SLOAD RVQ  
ONE/SP

ONE/SP DEC 1

1412THE

# P20-P25

PAGE 567

|    |      |      |             |                                     |    |
|----|------|------|-------------|-------------------------------------|----|
| 1  | FHNM | 2DEC | 740800 B-20 | # 400 NAUTICAL MILES IN METERS B-20 | 1  |
| 2  |      |      |             |                                     | 2  |
| 3  |      |      |             |                                     | 3  |
| 4  |      |      |             |                                     | 4  |
| 5  |      |      |             |                                     | 5  |
| 6  |      |      |             |                                     | 6  |
| 7  |      |      |             |                                     | 7  |
| 8  |      |      |             |                                     | 8  |
| 9  |      |      |             |                                     | 9  |
| 10 |      |      |             |                                     | 10 |
| 11 |      |      |             |                                     | 11 |
| 12 |      |      |             |                                     | 12 |
| 13 |      |      |             |                                     | 13 |
| 14 |      |      |             |                                     | 14 |
| 15 |      |      |             |                                     | 15 |
| 16 |      |      |             |                                     | 16 |
| 17 |      |      |             |                                     | 17 |
| 18 |      |      |             |                                     | 18 |
| 19 |      |      |             |                                     | 19 |
| 20 |      |      |             |                                     | 20 |
| 21 |      |      |             |                                     | 21 |
| 22 |      |      |             |                                     | 22 |
| 23 |      |      |             |                                     | 23 |
| 24 |      |      |             |                                     | 24 |
| 25 |      |      |             |                                     | 25 |
| 26 |      |      |             |                                     | 26 |
| 27 |      |      |             |                                     | 27 |
| 28 |      |      |             |                                     | 28 |
| 29 |      |      |             |                                     | 29 |
| 30 |      |      |             |                                     | 30 |
| 31 |      |      |             |                                     | 31 |
| 32 |      |      |             |                                     | 32 |
| 33 |      |      |             |                                     | 33 |
| 34 |      |      |             |                                     | 34 |
| 35 |      |      |             |                                     | 35 |
| 36 |      |      |             |                                     | 36 |
| 37 |      |      |             |                                     | 37 |
| 38 |      |      |             |                                     | 38 |
| 39 |      |      |             |                                     | 39 |
| 40 |      |      |             |                                     | 40 |
| 41 |      |      |             |                                     | 41 |
| 42 |      |      |             |                                     | 42 |
| 43 |      |      |             |                                     | 43 |
| 44 |      |      |             |                                     | 44 |
| 45 |      |      |             |                                     | 45 |
| 46 |      |      |             |                                     | 46 |
| 47 |      |      |             |                                     | 47 |
| 48 |      |      |             |                                     | 48 |
| 49 |      |      |             |                                     | 49 |
| 50 |      |      |             |                                     | 50 |
| 51 |      |      |             |                                     | 51 |
| 52 |      |      |             |                                     | 52 |
| 53 |      |      |             |                                     | 53 |
| 54 |      |      |             |                                     | 54 |
| 55 |      |      |             |                                     | 55 |
| 56 |      |      |             |                                     | 56 |
| 57 |      |      |             |                                     | 57 |
| 58 |      |      |             |                                     | 58 |
| 59 |      |      |             |                                     | 59 |
| 60 |      |      |             |                                     | 60 |

1412THE



```
1 # PROGRAM NAME: LRS22.1 (DATA READ SUBROUTINE 1)
2 # MOD. NO.: 1 BY: P. VOLANTE SDC DATE: 11-15-66
3
4 #
5 # FUNCTIONAL DESCRIPTION:
6 # 1) READS RENDEZVOUS RADAR RANGE AND RANGE-RATE, TRUNNION AND SHAFT ANGLES, THREE CDU VALUES AND TIME. CONVERTS THIS
7 # DATA AND LEAVES IT FOR THE MEASUREMENT INCORPORATION ROUTINE (LSR22.3). CHECKS FOR THE RR DATA GOOD DISCRETE, FOR
8 # RR REPOSITION AND RR CDU FAIL
9 # 2) COMPARES RADAR LOS WITH LOS COMPUTED FROM STATE VECTORS TO SEE IF THEY ARE WITHIN THREE DEGREES
10
11 # CALLING SEQUENCE: BANKCALL FOR LRS22.1
12 #
13 # SUBROUTINES CALLED:
14 # RRDOT LPS20.1
15 # RRRANGE BANKCALL
16 # RADSTALL CDULOGIC
17 # RRNB SMNB
18 #
19 # NORMAL EXIT: RETURN TO CALLER+1 WITH MPAC SET TO +0
20 #
21 # ERROR EXITS: RETURN TO CALLER+1 WITH ERROR CODE STORED IN MPAC AS FOLLOWS:
22 # 00001 -- ERROR EXIT 1 -- RR DATA NO GOOD (NO RR DATA GOOD DISCRETE OR RR CDU FAIL OR RR REPOSITION)
23 # 00002 -- ERROR EXIT 2 -- RR LOS NOT WITHIN THREE DEGREES OF LOS COMPUTED FROM STATE VECTORS
24 #
25 # ALARMS: 521 -- COULD NOT READ RADAR DATA (RR DATA GOOD DISCRETE NOT PRESENT BEFORE AND AFTER READING THE RADAR)
26 # (THIS ALARM IS ISSUED BY RADARREAD SUBROUTINE WHICH IS ENTERED FROM A RADARUPT)
27 #
28 # OUTPUT: RRLOSVEC -- THE RR LINE-OF-SIGHT VECTOR (USED BY LRS22.2) -- A HALF-UNIT VECTOR
29 # RM -- THE RR RANGE READING (TO THE CSM) DP, IN METERS SCALED BY B-29 (USED BY LRS22.2 AND LRS22.3)
30 #
31 # ALL OF THE FOLLOWING OUTPUTS ARE USED BY LRS22.3:
32 # RDOTM -- THE RR RANGE-RATE READING, DP, IN METERS PER CENTISECOND, SCALED BY B-7
33 # RRTRUN -- THE RR TRUNNION ANGLE, DP, IN REVOLUTIONS, SCALED B0
34 # RRSHAFT -- RR SHAFT ANGLE, DP, IN REVOLUTIONS, SCALED B0
35 # AIG,AMG,AOG -- THE CDU ANGLES, THREE SP WORDS
36 # MKTIME -- THE TIME OF THE RR READING, DP, IN CENTISECONDS
37 #
38 # ERASABLE INITIALIZATION REQUIRED:
39 # RNRAD, THE RADAR READ COUNTER FROM WHICH IS OBTAINED:
```

```
1) RR RANGE SCALED 9.38 FT. PER BIT ON THE LOW SCALE AND 75.04 FT. PER BIT ON THE HIGH SCALE
2) RR RANGE RATE, SCALED .6278 FT./SEC. PER BIT
THE CDU ANGLES FROM CDUX, CDUY, CDUZ, AND TIME1 AND TIME2
#
DEBRIS: LRS22.1X, A, L, Q, PUSHLIST
```

```
BANK 32
SETLOC LRS22
```

```
BANK
COUNT* $$/LRS22
```

```
LRS22.1 TC MAKECADR
 TS LRS22.1X
 TC DOWNFLAG
```

```
ADRES RNGSCFLG
INHINT
CAF BIT3
```

```
EXTEND # GET RR RANGE SCALE
RAND CHAN33 # FROM CHANNEL 33 BIT 3
TS L
```

```
CS RRRSBIT
MASK RADMODES
AD L
```

```
TS RADMODES
RELINT
READRDOT TC BANKCALL
```

```
CADR RRRDOT # READ RANGE-RATE (ONE SAMPLE)
TC BANKCALL
CADR RADSTALL # WAIT FOR DATA READ COMPLETION
TCF EREXIT1 # COULD NOT READ RADAR-ERROR EXIT 1
```

```
INHINT # NO INTERRUPTS WHILE READING TIME AND CDU
DXCH TIMEHOLD # SET MARK TIME EQUAL TO THE MID-POINT
DXCH MPAC +5 # TEMP BUFFER FOR DOWNLINK
DXCH SAMPLSUM # SAVE RANGE-RATE READING
```

```
DXCH RDOTMSAV
EXTEND
```

```
DCA CDUY # SAVE ICDU ANGLES
DXCH MPAC +3 # TEMP BUFFER FOR DOWNLINK
CA CDUX
TS MPAC +2 # TEMP BUFFER FOR DOWNLINK
```

```
EXTEND
DCA TIME2 # SAVE TIME
DXCH MPAC # SAVE TIME OF CDUY READINGS IN MPAC
```

```
EXTEND
DCA CDUT # SAVE TRUNNION AND SHAFT ANGLES FOR RRNB
DXCH TANG
```

|        |            |                                            |  |
|--------|------------|--------------------------------------------|--|
| RELINT |            |                                            |  |
| TC     | BANKCALL   |                                            |  |
| CADR   | RRRANGE    | # READ RR RANGE (ONE SAMPLE)               |  |
| TC     | BANKCALL   |                                            |  |
| CADR   | RADSTALL   | # WAIT FOR READ COMPLETE                   |  |
| TC     | CHEXERR    | # CHECK FOR ERRORS DURING READ             |  |
| INHINT |            | # COPY CYCLE FOR MARK DATA ON DOWNLINK     |  |
| DXCH   | DNRRANGE   | # RANGE, RANGE RATE (RAW DATA)             |  |
| DXCH   | RANGRDOT   |                                            |  |
| DXCH   | MPAC +5    |                                            |  |
| DXCH   | MKTIME     | # MARK TIME                                |  |
| DXCH   | MPAC +3    |                                            |  |
| DXCH   | AIG        | # CDUY, CDUZ                               |  |
| EXTEND |            |                                            |  |
| DCA    | TANG       | # PRESERVE TANG                            |  |
| DXCH   | TANGNB     | # TRUNNION AND SHAFT ANGLES                |  |
| CA     | MPAC +2    |                                            |  |
| TS     | AOG        | # CDUX                                     |  |
| TC     | INTPRET    |                                            |  |
| STODL  | 20D        | # SAVE TIME OF CDU READINGS IN 20D         |  |
|        | RDOTMSAV   | # CONVERT RDOT UNITS AND SCALING           |  |
| SL     | DMPR       | # START WITH READING SCALED B-28, -.6278   |  |
|        | 14D        | # FT./SECOND PER BIT                       |  |
|        | RDOTCONV   | # END WITH METERS/CENTISECOND, B-7         |  |
| STORE  | RDOTM      |                                            |  |
| SLOAD  | RTB        |                                            |  |
|        | TANG       | # GET TRUNNION ANGLE                       |  |
|        | CDULOGIC   | # CONVERT TO DP ONES COMP. IN REVOLUTIONS  |  |
| STORE  | RRTRUN     | # AND SAVE FOR TMI ROUTINE (LSR22.3)       |  |
| SLOAD  | RTB        |                                            |  |
|        | TANG +1    | # DITTO FOR SHAFT ANGLE                    |  |
|        | CDULOGIC   |                                            |  |
| STODL  | RRSHAFT    |                                            |  |
|        | SAMPLSUM   |                                            |  |
| DMP    | SL2R       | # CONVERT UNITS AND SCALING DP RANGE       |  |
|        | RANGCONV   | # PER BIT, END WITH METERS, SCALED -29     |  |
| STCALL | RM         |                                            |  |
|        | RRNB       | # COMPUTE RADAR LOS USING RRNB             |  |
| STODL  | RRBORSIT   | # AND SAVE                                 |  |
|        | 20D        |                                            |  |
| STCALL | TDEC1      | # GET STATE VECTOR LOS AT TIME OF CDU READ |  |
|        | LPS20.1    |                                            |  |
| EXIT   |            |                                            |  |
| CA     | AIG        | # STORE IMU CDU ANGLES AT MARKTIME         |  |
| TS     | CDUSPOT    | # IN CDUSPOT FOR TRG*SMNB                  |  |
| CA     | AMG        |                                            |  |
| TS     | CDUSPOT +2 |                                            |  |
| CA     | AOG        |                                            |  |
| TS     | CDUSPOT +4 |                                            |  |
| TC     | INTPRET    |                                            |  |

|    |    |
|----|----|
| 1  | 2  |
| 3  | 4  |
| 5  | 6  |
| 7  | 8  |
| 9  | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |
| 35 | 36 |
| 37 | 38 |
| 39 | 40 |
| 41 | 42 |
| 43 | 44 |
| 45 | 46 |
| 47 | 48 |
| 49 | 50 |
| 51 | 52 |
| 53 | 54 |
| 55 | 56 |
| 57 | 58 |
| 59 | 60 |
| 61 | 62 |
| 63 | 64 |
| 65 | 66 |
| 67 | 68 |
| 69 | 70 |
| 71 | 72 |
| 73 | 74 |
| 75 | 76 |
| 77 | 78 |
| 79 | 80 |

```
1 # PROGRAM NAME -- LRS22.2 (DATA READ SUBROUTINE 2)
2 # MOD. NO.: 1 BY: P. VOLANTE SDC DATE: 4-11-67
3
4 #
5 # FUNCTIONAL DESCRIPTION:
6 # (YES, I KNOW POINT #1 IS MISSING. IT IS MISSING FROM THE PROGRAM LISTING -- RSB 2003)
7 # 2) CHECKS IF THE RR LOS (I.E., THE RADAR BORESIGHT VECTOR) IS WITHIN 30 DEGREES OF THE LM +Z AXIS
8 #
9 # CALLING SEQUENCE: BANKCALL FOR LRS22.2
10 #
11 # SUBROUTINES CALLED: G+N, AUTO, SETMAXDB
12 #
13 # NORMAL EXIT: RETURN TO CALLER WITH MPAC SET TO +0 (VIA SWRETURN)
14 #
15 # ERROR EXIT: RETURN TO CALLER WITH MPAC SET TO 00001 -- RADAR LOS NOT WITHIN 30 DEGREES OF LM +Z AXIS.
16 #
17 # ALARMS: NONE
18 #
19 # ERASABLE INITIALIZATION REQUIRED:
20 # RRLOSVEC -- THE RR LINE-OF-SIGHT VECTOR -- A HALF UNIT VECTOR COMPUTED BY LRS22.1
21 # RM -- RR RANGE, METERS B-29, FROM LRS22.1
22 # BIT 14 CHANNEL 31 -- INDICATES AUTOPILOT IS IN AUTO MODE
23 #
24 # DEBRIS -- A,L,Q,MPAC -- PUSHLIST AND PUSHLOC ARE NOT CHANGED BY THIS ROUTINE
```

|          |        |             |                                         |
|----------|--------|-------------|-----------------------------------------|
|          | SETLOC | P20S        |                                         |
|          | BANK   |             |                                         |
| LRS22.2  | TC     | MAKECADR    |                                         |
|          | TS     | LRS22.1X    |                                         |
|          | TC     | INTPRET     |                                         |
| 30DEGCHK | DLOAD  | ACOS        | # CHECK IF RR LOS IS WITHIN 30 DEG OF   |
|          |        | RRBORSIT +4 | # THE SPACECRAFT +Z AXIT                |
|          |        |             | # BY TAKING ARCCOS OF Z-COMP. OF THE RR |
|          |        |             | # LOS VECTOR, A HALF UNIT VECTOR        |
|          |        |             | # IN NAV BASE AXES)                     |
|          | DSU    | BMN         |                                         |
|          |        | 30DEG       |                                         |
|          |        | OKEXIT      | # NORMAL EXIT -- WITHIN 30 DEG.         |
|          | EXIT   |             | # ERROR EXIT -- NOT WITHIN 30 DEG.      |
|          | CAF    | BIT1        | # SETS ERROR CODE IN MPAC               |
|          | TS     | MPAC        |                                         |
|          | TCF    | OUT22.2     |                                         |
| OKEXIT   | EXIT   |             | # NORMAL EXIT -- SET MPAC = ZERO        |



|    |         |      |            |                                   |
|----|---------|------|------------|-----------------------------------|
| 1  |         |      |            |                                   |
| 2  |         | CAF  | ZERO       |                                   |
| 3  |         | TS   | MPAC       |                                   |
| 4  | OUT22.2 | CAE  | LRS22.1X   |                                   |
| 5  |         | TC   | BANKJUMP   |                                   |
| 6  |         |      |            |                                   |
| 7  | 30DEG   | 2DEC | .083333333 | # THIRTY DEGREES, SCALED REVS, B0 |
| 8  |         |      |            |                                   |
| 9  |         |      |            |                                   |
| 10 |         |      |            |                                   |
| 11 |         |      |            |                                   |
| 12 |         |      |            |                                   |
| 13 |         |      |            |                                   |
| 14 |         |      |            |                                   |
| 15 |         |      |            |                                   |
| 16 |         |      |            |                                   |
| 17 |         |      |            |                                   |
| 18 |         |      |            |                                   |
| 19 |         |      |            |                                   |
| 20 |         |      |            |                                   |
| 21 |         |      |            |                                   |
| 22 |         |      |            |                                   |
| 23 |         |      |            |                                   |
| 24 |         |      |            |                                   |
| 25 |         |      |            |                                   |
| 26 |         |      |            |                                   |
| 27 |         |      |            |                                   |
| 28 |         |      |            |                                   |
| 29 |         |      |            |                                   |
| 30 |         |      |            |                                   |
| 31 |         |      |            |                                   |
| 32 |         |      |            |                                   |
| 33 |         |      |            |                                   |
| 34 |         |      |            |                                   |
| 35 |         |      |            |                                   |
| 36 |         |      |            |                                   |
| 37 |         |      |            |                                   |
| 38 |         |      |            |                                   |
| 39 |         |      |            |                                   |
| 40 |         |      |            |                                   |
| 41 |         |      |            |                                   |
| 42 |         |      |            |                                   |
| 43 |         |      |            |                                   |
| 44 |         |      |            |                                   |
| 45 |         |      |            |                                   |
| 46 |         |      |            |                                   |
| 47 |         |      |            |                                   |
| 48 |         |      |            |                                   |
| 49 |         |      |            |                                   |
| 50 |         |      |            |                                   |
| 51 |         |      |            |                                   |
| 52 |         |      |            |                                   |
| 53 |         |      |            |                                   |
| 54 |         |      |            |                                   |
| 55 |         |      |            |                                   |
| 56 |         |      |            |                                   |
| 57 |         |      |            |                                   |
| 58 |         |      |            |                                   |
| 59 |         |      |            |                                   |
| 60 |         |      |            |                                   |

```
1 # PROGRAM NAME -- LSR22.3 DATE -- 29 MAY 1967
2 # MOD. NO 3 LOG SECTION -- P20-P25
3 ASSEMBLY LEMP20S REV 10
4 # MOD. BY -- DANFORTH
5 #
6 # FUNCTIONAL DESCRIPTION:
7 # THIS ROUTINE COMPUTES THE B-VECTORS ADN DELTA Q FOR EACH OF THE QUANTITIES MEASURED BY THE RENDEZVOUS
8 # RADAR. (RANGE, RANGE RATE, SHAFT AND TRUNNION ANGLES). THE ROUTINE CALLS THE INCORP1 AND INCORP2 ROUTINES
9 # WHICH COMPUTE THE DEVIATIONS AND CORRECT THE STATE VECTOR.
10 #
11 # CALLING SEQUENCE:
12 # THIS ROUTINE IS PART OF P20 RENDEZVOUS NAVIGATION FOR THE LM COMPUTER ONLY. THE ROUTINE IS ENTERED FROM
13 # R22 LEM ONLY AND RETURNS DIRECTLY TO R22LEM FOLLOWING SUCCESSFUL INCORPORATION OF MEASURED DATA. IF THE
14 # COMPUTED STATE VECTOR DEVIATIONS EXCEED THE MAXIMUM PERMITTED. THE ROUTINE RETURNS TO R22LEM TO DISPLAY
15 # THE DEVIATIONS. IF THE ASTRONAUT ACCEPTS THE DATA R22LEM RETURNS TO LSR22.3 TO INCORPORATE THE
16 # DEVIATIONS INTO THE STATE VECTOR. IF THE ASTRONAUT REJECTS THE DEVIATIONS, NO MORE MEASUREMENTS ARE
17 # PROCESSED FOR THIS MARK, I.E., R22LEM GETS THE NEXT MARK.
18 #
19 # SUBROUTINES CALLED:
20 # WLIMIT LGCUPDTE INTEGRV INCORP1 ARCTAN
21 # GETULC RADARANG INCORP2 NBSM INTSTALL
22 #
23 # OUTPUT:
24 # CORRECTED LM OR CSM STATE VECTOR (PERMANENT)
25 # NUMBER OF MARKS INCORPORATED IN MARKCTR
26 # MAGNITUDE OF POSITION DEVIATION (FOR DISPLAY) IN R22DISP METERS B-29
27 # MAGNITUDE OF VELOCITY DEVIATION (FOR DISPLAY) IN R22DISP +2 M/CSEC B-7
28 # UPDATED W-MATRIX
29 #
30 # ERASABLE INITIALIZATION REQUIRED:
31 # LM AND CSM STATE VECTORS
32 # W-MATRIX
33 # MARK TIME IN MKTIME
34 # RADAR RANGE IN RM METERS B-29
35 # RANGE RATE IN RDOTM METERS/CSES B-7
36 # SHAFT ANGLE IN RRSHAFT REVS. B0
37 # TRUNNION ANGLE IN RRTRUN REVS. B0
38 # GIMBAL ANGLES INNER IN AIG
39 # MIDDLE IN AMG
40 # OUTER IN ACG
41 # REFSMMAT
42 # RENDWFLG
43 # NOANGFLG
44 # VEHUPFLG
45 #
46 # DEBRIS:
47 # PUSHLIST -- ALL
48 # MX, MY, MZ (VECTORS)
```

# P20-P25

|    |         |                                                        |            |    |
|----|---------|--------------------------------------------------------|------------|----|
| 1  |         |                                                        |            | 1  |
| 2  | #       | ULC, RXZ, SINTHETA, LGRET, RDRET, BVECTOR, W.IND, X78T |            | 2  |
| 3  |         |                                                        |            | 3  |
| 4  |         | BANK                                                   | 13         | 4  |
| 5  |         | SETLOC                                                 | P20S3      | 5  |
| 6  |         | BANK                                                   |            | 6  |
| 7  |         |                                                        |            | 7  |
| 8  |         | EBANK=                                                 | LOSCOUNT   | 8  |
| 9  |         | COUNT*                                                 | \$\$/LSR22 | 9  |
| 10 | LSR22.3 | CALL                                                   |            | 10 |
| 11 |         | BON                                                    | GRP2PC     | 11 |
| 12 |         |                                                        | SET        | 12 |
| 13 |         |                                                        | SURFFLAG   | 13 |
| 14 |         |                                                        | LSR22.4    | 14 |
| 15 |         |                                                        | DMENFLG    | 15 |
| 16 |         | BOFF                                                   | CALL       | 16 |
| 17 |         |                                                        | VEHUPFLG   | 17 |
| 18 |         |                                                        | DOLEM      | 18 |
| 19 |         | CLEAR                                                  | INTSTALL   | 19 |
| 20 |         |                                                        | CALL       | 20 |
| 21 |         |                                                        | VINTFLAG   | 21 |
| 22 |         |                                                        | SETIFLGS   | 22 |
| 23 |         | CALL                                                   |            | 23 |
| 24 |         |                                                        | INTGRCAL   | 24 |
| 25 |         | CALL                                                   |            | 25 |
| 26 |         |                                                        | GRP2PC     | 26 |
| 27 |         | CALL                                                   |            | 27 |
| 28 |         |                                                        | INTSTALL   | 28 |
| 29 |         | CLEAR                                                  | BOFF       | 29 |
| 30 |         |                                                        | DIMOFLAG   | 30 |
| 31 |         |                                                        | RENDWFLG   | 31 |
| 32 |         |                                                        | NOTWCSM    | 32 |
| 33 |         | SET                                                    | SET        | 33 |
| 34 |         |                                                        |            | 34 |
| 35 |         |                                                        | DIMOFLAG   | 35 |
| 36 | NOTWCSM | SET                                                    | D6OR9FLG   | 36 |
| 37 |         |                                                        | CLEAR      | 37 |
| 38 |         |                                                        | VINTFLAG   | 38 |
| 39 |         | SET                                                    | INTYPFLG   | 39 |
| 40 |         |                                                        | CALL       | 40 |
| 41 |         |                                                        | STATEFLG   | 41 |
| 42 |         | GOTO                                                   | INTGRCAL   | 42 |
| 43 |         |                                                        | MARKTEST   | 43 |
| 44 | DOLEM   | CALL                                                   |            | 44 |
| 45 |         |                                                        | INTSTALL   | 45 |
| 46 |         | SET                                                    | CALL       | 46 |
| 47 |         |                                                        | VINTFLAG   | 47 |
| 48 |         |                                                        | SETIFLGS   | 48 |
| 49 |         | CALL                                                   |            | 49 |
| 50 |         |                                                        | INTGRCAL   | 50 |
| 51 |         |                                                        |            | 51 |
| 52 |         |                                                        |            | 52 |
| 53 |         |                                                        |            | 53 |
| 54 |         |                                                        |            | 54 |
| 55 |         |                                                        |            | 55 |
| 56 |         |                                                        |            | 56 |
| 57 |         |                                                        |            | 57 |
| 58 |         |                                                        |            | 58 |
| 59 |         |                                                        |            | 59 |
| 60 |         |                                                        |            | 60 |

1412THE  
1



|    |          |        |          |                                          |
|----|----------|--------|----------|------------------------------------------|
| 1  |          |        |          |                                          |
| 2  |          | CALL   |          |                                          |
| 3  |          |        | GRP2PC   |                                          |
| 4  |          | CALL   |          |                                          |
| 5  |          |        | INTSTALL |                                          |
| 6  |          | CLEAR  | BOFF     |                                          |
| 7  |          |        | DIM0FLAG |                                          |
| 8  |          |        | RENDWFLG |                                          |
| 9  |          |        | NOTWLEM  |                                          |
| 10 |          | SET    | SET      | # LM WITH W-MATRIX INTEGRATION           |
| 11 |          |        | DIM0FLAG |                                          |
| 12 |          |        | D6OR9FLG |                                          |
| 13 | NOTWLEM  | CLEAR  | CLEAR    |                                          |
| 14 |          |        | INTYPFLG |                                          |
| 15 |          |        | VINTFLAG |                                          |
| 16 |          | SET    | CALL     |                                          |
| 17 |          |        | STATEFLG |                                          |
| 18 |          |        | INTGRCAL |                                          |
| 19 | MARKTEST | BON    | CALL     | # HAS W-MATRIX BEEN INVALIDATED          |
| 20 |          |        | RENDWFLG | # HAS W-MATRIX BEEN INVALIDATED          |
| 21 |          |        | RANGEBQ  |                                          |
| 22 |          |        | WLINIT   | # YES -- REINITIALIZE                    |
| 23 | RANGEBQ  | BON    | EXIT     | # DON'T CALL R65 IF ON SURFACE           |
| 24 |          |        | SURFFLAG |                                          |
| 25 |          |        | RANGEBQ1 |                                          |
| 26 |          | CA     | ZERO     |                                          |
| 27 |          | TS     | R65CNTR  |                                          |
| 28 |          | TC     | BANKCALL |                                          |
| 29 |          | CADR   | R65LEM   |                                          |
| 30 |          | TC     | INTPRET  |                                          |
| 31 | RANGEBQ1 | AXT,2  | BON      | # CLEAR X2                               |
| 32 |          |        | 0        |                                          |
| 33 |          |        | LMOONFLG | # IS MOON SPHERE OF INFLUENCE            |
| 34 |          |        | SETX2    | # YES. STORE ZERO IN SCALSHFT REGISTER   |
| 35 |          | INCR,2 |          |                                          |
| 36 |          |        | 2        |                                          |
| 37 | SETX2    | SXA,2  | CALL     |                                          |
| 38 |          |        | SCALSHFT | # 0 -- MOON. 2 -- EARTH.                 |
| 39 |          |        | GRP2PC   |                                          |
| 40 |          | AXT,1  | SXA,1    | # STORE RANGE CODE (1) FOR R3 IN NOUN 49 |
| 41 |          |        | 1        |                                          |
| 42 |          |        | WHCHREAD |                                          |
| 43 |          | SLOAD  | SR       | # GET SINGLE PRECISION RVARMIN (B-12)    |
| 44 |          |        | RVARMIN  | # SHIFT TO TRIPLE PRECISION (B-40)       |
| 45 |          |        | 28D      |                                          |
| 46 |          | RTB    |          |                                          |
| 47 |          |        | TPMODE   | # AND SAVE IN 20D                        |
| 48 |          | STORE  | 20D      |                                          |
| 49 |          | CALL   |          | # BEGIN COMPUTING THE B-VECTORS, DELTAQ  |
| 50 |          |        | GETULC   | # B-VECTORS FOR RANGE                    |
| 51 |          | BON    | VCOMP    | # B0, COMP. IF LM BEING CORRECTED        |
| 52 |          |        |          |                                          |
| 53 |          |        |          |                                          |
| 54 |          |        |          |                                          |
| 55 |          |        |          |                                          |
| 56 |          |        |          |                                          |
| 57 |          |        |          |                                          |
| 58 |          |        |          |                                          |
| 59 |          |        |          |                                          |
| 60 |          |        |          |                                          |

|    |       |              |                                           |
|----|-------|--------------|-------------------------------------------|
| 1  |       | VEHUPFLG     |                                           |
| 2  |       | +1           |                                           |
| 3  |       |              |                                           |
| 4  | STOVL | BVECTOR      |                                           |
| 5  |       | ZEROVECS     |                                           |
| 6  | STORE | BVECTOR +6   | # B1                                      |
| 7  | STODL | BVECTOR +12D | # B2                                      |
| 8  |       | 36D          |                                           |
| 9  | SRR*  | BDSU         |                                           |
| 10 |       | 2,2          | # SHIFT FROM EARTH/MOON SPHERE TO B-29    |
| 11 |       | RM           | # RM - (MAGNITUDE RCSM-RLM)               |
| 12 | SLR*  |              |                                           |
| 13 |       | 2,2          | # SHIFT TO EARTH/MOON SPHERE              |
| 14 | STODL | DELTAQ       | # EARTH B-29. MOON B-27                   |
| 15 |       | 36D          | # RLC B-29/B-27                           |
| 16 | NORM  | DSQ          | # NORMALIZE AND SQUARE                    |
| 17 |       | X1           |                                           |
| 18 | DMP   | SR*          |                                           |
| 19 |       | RANGEVAR     | # MULTIPLY BY RANGEVAR (B12) THEN         |
| 20 |       | 0 -2,1       | # UNNORMALIZE                             |
| 21 | SR*   | SR*          |                                           |
| 22 |       | 0,1          |                                           |
| 23 |       | 0,2          |                                           |
| 24 | SR*   | RTB          |                                           |
| 25 |       | 0,2          |                                           |
| 26 |       | TPMODE       |                                           |
| 27 | STORE | VARIANCE     | # B-40                                    |
| 28 | DCOMP | TAD          |                                           |
| 29 |       | 20D          | # B-40                                    |
| 30 | BMN   | TLOAD        |                                           |
| 31 |       | QOK          |                                           |
| 32 |       | 20D          | # B-40                                    |
| 33 | STORE | VARIANCE     |                                           |
| 34 | QOK   | CALL         |                                           |
| 35 |       | LGCUPDTE     |                                           |
| 36 |       |              |                                           |
| 37 | SSP   | CALL         |                                           |
| 38 |       | WHCHREAD     |                                           |
| 39 | DEC   | 2            | # STORE R-RATE CODE (2) FOR R3 IN NOUN 49 |
| 40 |       | GRP2PC       |                                           |
| 41 | CALL  |              | # B-VECTOR, DELTAQ FOR RANGE RATE         |
| 42 |       | GETULC       |                                           |
| 43 | PDDL  | SR*          | # GET RLC SCALED B-29/B-27                |
| 44 |       | 36D          | # AND SHIFT TO B-23                       |
| 45 |       | 0 -4,2       |                                           |
| 46 | STOVL | 36D          | # THEN STORE BACK IN 36D                  |
| 47 | BON   | VCOMP        | # B1, COMP. IF LM BEING CORRECTED         |
| 48 |       | VEHUPFLG     |                                           |
| 49 |       | +1           |                                           |
| 50 | VXSC  |              |                                           |
| 51 |       | 36D          | # B1 = RLC (B-24/B-22)                    |
| 52 |       |              |                                           |
| 53 |       |              |                                           |
| 54 |       |              |                                           |
| 55 |       |              |                                           |
| 56 |       |              |                                           |
| 57 |       |              |                                           |
| 58 |       |              |                                           |
| 59 |       |              |                                           |
| 60 |       |              |                                           |

|    |     |       |              |                                        |    |
|----|-----|-------|--------------|----------------------------------------|----|
| 1  |     |       |              |                                        | 1  |
| 2  |     | STOVL | BVECTOR +6   |                                        | 2  |
| 3  |     |       | NUVLEM       |                                        | 3  |
| 4  |     | VSR*  | VAD          |                                        | 4  |
| 5  |     |       | 6,2          | # SHIFT FOR EARTH/MOON SPHERE          | 5  |
| 6  |     |       | VCVLEM       | # EARTH B-7. MOON B-5                  | 6  |
| 7  |     | PDVL  | VSR*         | # VL TO PD6                            | 7  |
| 8  |     |       | NUVCSM       |                                        | 8  |
| 9  |     |       | 6,2          | # SHIFT FOR EARTH/MOON SPHERE          | 9  |
| 10 |     | VAD   | VSU          |                                        | 10 |
| 11 |     |       | VCVCSM       |                                        | 11 |
| 12 |     | PDVL  | DOT          | # VC - VL = VLC TO PD6                 | 12 |
| 13 |     |       | 0            |                                        | 13 |
| 14 |     |       | 6            |                                        | 14 |
| 15 |     | PUSH  | SRR*         | # RDOT B-8/B-6 TO PD12                 | 15 |
| 16 |     |       | 2,2          | # SHIFT FROM EARTH/MOON SPHERE TO B-8  | 16 |
| 17 |     | DSQ   | DMPR         | # RDOT**2 B-16 X RATEVAR B12           | 17 |
| 18 |     |       | RATEVAR      |                                        | 18 |
| 19 |     | STORE | VARIANCE     |                                        | 19 |
| 20 |     | SLOAD | SR           |                                        | 20 |
| 21 |     |       | VVARMIN      | # GET SINGLE PRECISION VVARMIN (B+12)  | 21 |
| 22 |     |       | 16D          | # SHIFT TO DP (B-4)                    | 22 |
| 23 |     | STORE | 24D          | # AND SAVE IN 24D                      | 23 |
| 24 |     | DSU   | BMN          | # IS MIN. VARIANCE > COMPUTED VARIANCE | 24 |
| 25 |     |       | VARIANCE     |                                        | 25 |
| 26 |     |       | VOK          | # BRANCH -- NO                         | 26 |
| 27 |     | DLOAD |              | # YES -- USE MINIMUM VARIANCE          | 27 |
| 28 |     |       | 24D          |                                        | 28 |
| 29 |     | STORE | VARIANCE     |                                        | 29 |
| 30 | VOK | DLOAD | SR2          | # RDOT (PD12) FROM B-8/B-6             | 30 |
| 31 |     | PDDL  | SLR*         | # TO B-10/B-8                          | 31 |
| 32 |     |       | RDOTM        | # SHIFT TO EARTH/MOON SPHERE           | 32 |
| 33 |     |       | 0 -1,2       | # B-7 TO B-10/B-8                      | 33 |
| 34 |     | DSU   |              |                                        | 34 |
| 35 |     | DMPR  |              |                                        | 35 |
| 36 |     |       | 36D          |                                        | 36 |
| 37 |     | STOVL | DELTAQ       | # B-33                                 | 37 |
| 38 |     |       | 0            | # NOW GET B0                           | 38 |
| 39 |     | VXV   | VXV          | # (ULC X VLC) X ULC                    | 39 |
| 40 |     | BON   | VCOMP        | # B0, COMP. IF LM BEING CORRECTED      | 40 |
| 41 |     |       | VEHUPFLG     |                                        | 41 |
| 42 |     |       | +1           |                                        | 42 |
| 43 |     | VSR*  |              |                                        | 43 |
| 44 |     |       | 0 -2,2       | # SCALED B-5                           | 44 |
| 45 |     | STOVL | BVECTOR      |                                        | 45 |
| 46 |     |       | ZEROVECS     |                                        | 46 |
| 47 |     | STORE | 20D          | # ZERO OUT 20 TO 25 IN PUSHLIST        | 47 |
| 48 |     | STOVL | BVECTOR +12D |                                        | 48 |
| 49 |     |       | BVECTOR      |                                        | 49 |
| 50 |     | ABVAL | NORM         | # LOAD B0, GET MAGNITUDE AND NORMALIZE | 50 |
| 51 |     |       | 20D          | # SHIFT COUNT IN 20D                   | 51 |
| 52 |     |       |              |                                        | 52 |
| 53 |     |       |              |                                        | 53 |
| 54 |     |       |              |                                        | 54 |
| 55 |     |       |              |                                        | 55 |
| 56 |     |       |              |                                        | 56 |
| 57 |     |       |              |                                        | 57 |
| 58 |     |       |              |                                        | 58 |
| 59 |     |       |              |                                        | 59 |
| 60 |     |       |              |                                        | 60 |

|    |        |        |             |                                             |
|----|--------|--------|-------------|---------------------------------------------|
| 1  |        |        |             |                                             |
| 2  |        | VLOAD  | ABVAL       |                                             |
| 3  |        |        | BVECTOR +6D | # LOAD B1, GET MAGNITUDE AND NORMALIZE      |
| 4  |        | NORM   | DLOAD       |                                             |
| 5  |        |        | 22D         | # SHIFT COUNT IN 22D                        |
| 6  |        |        | 22D         | # FIND WHICH SHIFT IS SMALLER               |
| 7  |        | DSU    | BMN         | # BRANCH -- B0 HAS A SMALLER SHIFT COUNT    |
| 8  |        |        | 20D         |                                             |
| 9  |        |        | VOK1        |                                             |
| 10 |        | LXA,1  | GOTO        |                                             |
| 11 |        |        | 22D         | # LOAD X2 WITH THE SMALLER SHIFT COUNT      |
| 12 |        |        | VOK2        |                                             |
| 13 | VOK1   | LXA,1  |             |                                             |
| 14 |        |        | 20D         |                                             |
| 15 | VOK2   | VLOAD  | VSL*        | # THEN ADJUST B0, B1, DELTAQ AND VARIANCE   |
| 16 |        |        | BVECTOR     | # WITH THI SSHIFT COUNT                     |
| 17 |        |        | 0,1         |                                             |
| 18 |        | STOVL  | BVECTOR     |                                             |
| 19 |        |        | BVECTOR +6  |                                             |
| 20 |        | VSL*   |             |                                             |
| 21 |        |        | 0,1         |                                             |
| 22 |        | STODL  | BVECTOR +6  |                                             |
| 23 |        |        | DELTAQ      |                                             |
| 24 |        | SL*    |             |                                             |
| 25 |        |        | 0,1         |                                             |
| 26 |        | STORE  | DELTAQ      |                                             |
| 27 |        | DLOAD  | SL*         | # GET RLC AND ADJUST FOR SCALE SHIFT        |
| 28 |        |        | 36D         |                                             |
| 29 |        |        | 0 -1,1      |                                             |
| 30 |        | DSQ    | DMP         | # MULTIPLY RLC**2 BY VARIANCE               |
| 31 |        |        | VARIANCE    |                                             |
| 32 |        | SL4    | RTB         | # SHIFT TO CONFORM TO BVECTORS AND DELTAQ   |
| 33 |        |        | TPMODE      |                                             |
| 34 |        | STCALL | VARIANCE    | # AND STORE TP VARIANCE                     |
| 35 |        |        | LGCUPDTE    |                                             |
| 36 |        |        |             |                                             |
| 37 |        | CALL   |             |                                             |
| 38 |        |        | GRP2PC      |                                             |
| 39 |        | BON    | EXIT        | # ARE ANGLES TO BE DONE                     |
| 40 |        |        | SURFFLAG    |                                             |
| 41 |        |        | RENDEND     | # NO                                        |
| 42 |        | EBANK= | AIG         |                                             |
| 43 | MXMYMZ | CAF    | AIGBANK     |                                             |
| 44 |        | TS     | BBANK       |                                             |
| 45 |        | CA     | AIG         | # YES, COMPUTE MX, MY, MZ                   |
| 46 |        | TS     | CDUSPOT     |                                             |
| 47 |        | CA     | AMG         |                                             |
| 48 |        | TS     | CDUSPOT +2  |                                             |
| 49 |        | CA     | AOG         |                                             |
| 50 |        | TS     | CDUSPOT +4  | # GIMBAL ANGLES NOW IN CDUSPOT FOR TRG*NBSM |
| 51 |        | TC     | INTPRET     |                                             |
| 52 |        |        |             |                                             |
| 53 |        |        |             |                                             |
| 54 |        |        |             |                                             |
| 55 |        |        |             |                                             |
| 56 |        |        |             |                                             |
| 57 |        |        |             |                                             |
| 58 |        |        |             |                                             |
| 59 |        |        |             |                                             |
| 60 |        |        |             |                                             |

|    |         |        |          |                                           |  |
|----|---------|--------|----------|-------------------------------------------|--|
| 1  |         |        |          |                                           |  |
| 2  |         | VLOAD  | CALL     |                                           |  |
| 3  |         |        | UNITX    |                                           |  |
| 4  |         |        | TRG*NBSM |                                           |  |
| 5  |         | VXM    | VSL1     |                                           |  |
| 6  |         |        | REFSMMAT |                                           |  |
| 7  |         | STOVL  | MX       |                                           |  |
| 8  |         |        | UNITY    |                                           |  |
| 9  |         | CALL   |          |                                           |  |
| 10 |         |        | *NBSM*   |                                           |  |
| 11 |         | VXM    | VSL1     |                                           |  |
| 12 |         |        | REFSMMAT |                                           |  |
| 13 |         | STOVL  | MY       |                                           |  |
| 14 |         |        | UNITZ    |                                           |  |
| 15 |         | CALL   |          |                                           |  |
| 16 |         |        | *NBSM*   |                                           |  |
| 17 |         | VXM    | VSL1     |                                           |  |
| 18 |         |        | REFSMMAT |                                           |  |
| 19 | SHAFTBQ | STCALL | MZ       |                                           |  |
| 20 |         |        | RADARANG |                                           |  |
| 21 |         | SSP    | CALL     | # STORE SHAFT CODE (3) FOR R3 IN NOUN 49  |  |
| 22 |         |        | WHCHREAD |                                           |  |
| 23 |         | DEC    | 3        |                                           |  |
| 24 |         |        | GRP2PC   |                                           |  |
| 25 |         | VLOAD  | DOT      | # COMPUTE DELTAQ,B VECTORS FOR SHAFT ANG. |  |
| 26 |         |        | ULC      |                                           |  |
| 27 |         |        | MX       |                                           |  |
| 28 |         | SL1    |          |                                           |  |
| 29 |         | STOVL  | SINTH    | # 18D                                     |  |
| 30 |         |        | ULC      |                                           |  |
| 31 |         | DOT    | SL1      |                                           |  |
| 32 |         |        | MZ       |                                           |  |
| 33 |         | STCALL | COSTH    | # 16D                                     |  |
| 34 |         |        | ARCTAN   |                                           |  |
| 35 |         | BDSU   | DMP      |                                           |  |
| 36 |         |        | RRSHAFT  |                                           |  |
| 37 |         |        | 2PI/8    |                                           |  |
| 38 |         | SL3R   | PUSH     |                                           |  |
| 39 |         | DLOAD  | SL3      |                                           |  |
| 40 |         |        | X789     |                                           |  |
| 41 |         | SRR*   | BDSU     | # SHIFT FROM -5/-3 TO B0                  |  |
| 42 |         |        | 0,2      |                                           |  |
| 43 |         | DMP    | SRR*     |                                           |  |
| 44 |         |        | RXZ      |                                           |  |
| 45 |         |        | 0,1      | # SHIFT TO EARTH/MOON SPHERE              |  |
| 46 |         | STOVL  | DELTAQ   | # EARTH B-29. MOON B-27                   |  |
| 47 |         |        | ULC      |                                           |  |
| 48 |         | VXV    | VSL1     |                                           |  |
| 49 |         |        | MY       |                                           |  |
| 50 |         | UNIT   |          |                                           |  |
| 51 |         | BOFF   | VCOMP    | # B0, COMP. IF CSM BEING CORRECTED        |  |
| 52 |         |        |          |                                           |  |
| 53 |         |        |          |                                           |  |
| 54 |         |        |          |                                           |  |
| 55 |         |        |          |                                           |  |
| 56 |         |        |          |                                           |  |
| 57 |         |        |          |                                           |  |
| 58 |         |        |          |                                           |  |
| 59 |         |        |          |                                           |  |
| 60 |         |        |          |                                           |  |

|    |           |                                              |    |
|----|-----------|----------------------------------------------|----|
| 1  | # FLE FLE |                                              | 1  |
| 2  |           | VEHUPFLG                                     | 2  |
| 3  |           | +1                                           | 3  |
| 4  |           |                                              | 4  |
| 5  | STOVL     | BVECTOR                                      | 5  |
| 6  |           | ZEROVECS                                     | 6  |
| 7  | STORE     | BVECTOR +6                                   | 7  |
| 8  | STODL     | BVECTOR +12D                                 | 8  |
| 9  |           | RXZ                                          | 9  |
| 10 | SR*       | SRR* # SHIFT FROM EARTH/MOON SPHERE TO B-25  | 10 |
| 11 |           | 0 -2,1                                       | 11 |
| 12 |           | 0,2                                          | 12 |
| 13 | STORE     | BVECTOR +12D                                 | 13 |
| 14 | SLOAD     |                                              | 14 |
| 15 |           | SHAFTVAR                                     | 15 |
| 16 | DAD       | DMP                                          | 16 |
| 17 |           | IMUVAR # RAD**2 B12                          | 17 |
| 18 |           | RXZ                                          | 18 |
| 19 | SRR*      | DMP                                          | 19 |
| 20 |           | 0,1 # SHIFT TO EARTH/MOON SPHERE             | 20 |
| 21 |           | RXZ                                          | 21 |
| 22 | SR*       | SR*                                          | 22 |
| 23 |           | 0 -2,1                                       | 23 |
| 24 |           | 0,2                                          | 24 |
| 25 | SR*       | RTB                                          | 25 |
| 26 |           | 0,2                                          | 26 |
| 27 | STCALL    | TPMODE # STORE VARIANCE TRIPLE PRECISION     | 27 |
| 28 |           | VARIANCE # B-40                              | 28 |
| 29 |           | LGCUPDTE                                     | 29 |
| 30 | CALL      |                                              | 30 |
| 31 |           | GRP2PC                                       | 31 |
| 32 | TRUNBQ    | CALL                                         | 32 |
| 33 |           | RADARANG                                     | 33 |
| 34 | SSP       | CALL # STORE TRUNNION CODE (4) FOR R3 IN N49 | 34 |
| 35 |           | WHCHREAD                                     | 35 |
| 36 | DEC       | 4                                            | 36 |
| 37 |           | GRP2PC                                       | 37 |
| 38 | VLOAD     | VXV                                          | 38 |
| 39 |           | ULC                                          | 39 |
| 40 |           | MY                                           | 40 |
| 41 | VSL1      | VXV                                          | 41 |
| 42 |           | ULC                                          | 42 |
| 43 | VSL1      | # (ULC X MY) X ULC                           | 43 |
| 44 | BOFF      | VCOMP # B0, COMP. IF CSM BEING CORRECTED     | 44 |
| 45 |           | VEHUPFLG                                     | 45 |
| 46 |           | +1                                           | 46 |
| 47 | STOVL     | BVECTOR                                      | 47 |
| 48 |           | ZEROVECS                                     | 48 |
| 49 | STORE     | BVECTOR +6                                   | 49 |
| 50 | STODL     | BVECTOR +12D                                 | 50 |
| 51 |           | RXZ                                          | 51 |
| 52 |           |                                              | 52 |
| 53 |           |                                              | 53 |
| 54 |           |                                              | 54 |
| 55 |           |                                              | 55 |
| 56 |           |                                              | 56 |
| 57 |           |                                              | 57 |
| 58 |           |                                              | 58 |
| 59 |           |                                              | 59 |
| 60 |           |                                              | 60 |

SR\*

# SHIFT FROM EARTH/MOON SPHERE TO B-25

0,2

TRUNVAR  
DMP  
IMUVAR

RXZ  
DMP  
0,1

```
SHIFT TO EARTH/MOON SPHERE
```

|     |   |
|-----|---|
| RXZ |   |
| SR* |   |
| 0   | — |

0,2  
RTB  
0,2

TPMODE  
VARIANCE  
SINTHETA

```
STORE VARIANCE TRIPLE PRECISION
```

BDS  
RRT  
SL3

## # SIN THETA IN PD6

2PI/8  
SL3  
X789 +2

BDS  
0,2  
SRR

## # SHIFT FROM -5/-3 TO B0

RXZ  
0,1  
DEL

# EARTH B-29. MOON B-27

LGCUPDTE

GRP2PC

GOTO

R22LEM93

```
LSR22.4 IS THE ENTRY TO PERFORM LUNAR SURFACE NAVIGATION FOR THE LM
COMPUTER ONLY. THIS ROUTINE COMPUTES THE BE-VECTORS AND DELTA Q FOR RANGE
AND RANGE RATE MEASURED BY THE RENDEZVOUS RADAR
```

|   |          |          |        |         |
|---|----------|----------|--------|---------|
| # | INTSTALL | LGCUPDTE | INCRP1 | RP-TO-R |
| # | INTEGRV  | GETULC   | INCRP2 |         |

```
CORRECTED CSM STATE VECTOR (PERMANENT)
NUMBER OF MARKS INCORPORATED IN MARKCTR
```

```
1
2 # MAGNITUDE OF POSITION DEVIATION (FOR DISPLAY) IN R22 DISP METERS B-29
3 # MAGNITUDE OF VELOCITY DEVIATION (FOR DISPLAY) IN R22DISP +2 M/CSEC B-7
4 # UPDATED W-MATRIX
5
6 # ERASABLE INITIALIZATION REQUIRED
7 # LM AND CSM STATE VECTORS
8 # W-MATRIX
9 # MARK TIME IN MKTIME
10 # RADAR RANGE IN RM METERS B-29
11 # RANGE RATE IN RDOTM METERS/CSEC B-7
12 # VEHUPFLG
```

```
13
14 LSR22.4 CALL
15 INTSTALL
16 SET CLEAR
17 STATEFLG
18 VINTFLAG # CALL TO GET LM POS + VEL IN REF COORD.
19
20 CALL
21 INTGRCAL
22
23 CLEAR GRP2PC
24 CALL
25 DMENFLG # SET MATRIX SIZE TO 6X6 FOR INCORP
26
27 DLOAD INTSTALL
28 BHIZ # IS THIS FIRST TIME THROUGH
29 MARKCTR
30
31 CLEAR INITWMX6 # YES, INITIALIZE 6X6 W-MATRIX
32 SET
33 D6OR9FLG
34
35 SET DIMOFLAG
36 CLEAR
37 VINTFLAG
38 INTYPFLG
39
40 CALL
41 INTGRCAL
42
43 GOTO
44 RANGEBQ
```

```
45
46 INITWMX6 CALL
47 WLIMIT # INITIALIZE W-MATRIX
48 SET CALL
49 VINTFLAG
50 SETIFLGS
51
52 CALL
53 INTGRCAL
54 GOTO
55 RANGEBQ
```

```
56
57 # THIS ROUTINE CLEARS RFINAL (DP) AND CALLS INTEGRV
58
59
60
```



|    |                                                                       |        |                                        |    |
|----|-----------------------------------------------------------------------|--------|----------------------------------------|----|
| 1  |                                                                       |        |                                        | 1  |
| 2  | INTGRCAL                                                              | STQ    | DLOAD                                  | 2  |
| 3  |                                                                       |        | IGRET                                  | 3  |
| 4  |                                                                       |        | MKTIME                                 | 4  |
| 5  |                                                                       | STCALL | TDEC1                                  | 5  |
| 6  |                                                                       |        | INTEGRV                                | 6  |
| 7  |                                                                       | GOTO   |                                        | 7  |
| 8  |                                                                       |        | IGRET                                  | 8  |
| 9  |                                                                       |        |                                        | 9  |
| 10 | # THIS ROUTINE INITIALIZES THE W-MATRIX BY ZEROING ALL W THEN SETTING |        |                                        | 10 |
| 11 | # DIAGONAL ELEMENTS TO INITIAL STORED VALUES.                         |        |                                        | 11 |
| 12 |                                                                       |        |                                        | 12 |
| 13 | WLIMIT                                                                | EBANK= | W                                      | 13 |
| 14 |                                                                       | EXIT   |                                        | 14 |
| 15 |                                                                       | CAF    | WBANK                                  | 15 |
| 16 |                                                                       | TS     | BBANK                                  | 16 |
| 17 |                                                                       | CAF    | WSIZE                                  | 17 |
| 18 |                                                                       | TS     | W.IND                                  | 18 |
| 19 |                                                                       | CAF    | ZERO                                   | 19 |
| 20 |                                                                       | INDEX  | W.IND                                  | 20 |
| 21 |                                                                       | TS     | W                                      | 21 |
| 22 |                                                                       | CCS    | W.IND                                  | 22 |
| 23 |                                                                       | TC     | -5                                     | 23 |
| 24 |                                                                       | CAF    | AIGBANK                                | 24 |
| 25 |                                                                       |        | # RESTORE EBANK 7                      | 25 |
| 26 |                                                                       | TS     | BBANK                                  | 26 |
| 27 |                                                                       | TC     | INTPRET                                | 27 |
| 28 |                                                                       | BON    | SLOAD                                  | 28 |
| 29 |                                                                       |        | # IF ON LUNAR SURFACE, INITIALIZE WITH | 29 |
| 30 |                                                                       |        | # WSURFPOS AND WSURFVEL INSTEAD OF     | 30 |
| 31 |                                                                       |        | # WRENDPOS AND WRENDVEL                | 31 |
| 32 |                                                                       | GOTO   |                                        | 32 |
| 33 | WLSRFPOS                                                              |        | WPOSTORE                               | 33 |
| 34 |                                                                       | SLOAD  |                                        | 34 |
| 35 | WPOSTORE                                                              | SR     | WSURFPOS                               | 35 |
| 36 |                                                                       |        | # SHIFT TO B-19 SCALE                  | 36 |
| 37 |                                                                       |        | 5                                      | 37 |
| 38 |                                                                       | STORE  | W                                      | 38 |
| 39 |                                                                       | STORE  | W +8D                                  | 39 |
| 40 |                                                                       | STORE  | W +16D                                 | 40 |
| 41 |                                                                       | BON    | SLOAD                                  | 41 |
| 42 |                                                                       |        | SURFFLAG                               | 42 |
| 43 |                                                                       |        | WLSRFVEL                               | 43 |
| 44 |                                                                       | GOTO   | WRENDVEL                               | 44 |
| 45 |                                                                       |        |                                        | 45 |
| 46 | WLSRFVEL                                                              |        | WVELSTOR                               | 46 |
| 47 |                                                                       | SLOAD  |                                        | 47 |
| 48 | WVELSTOR                                                              |        | WSURFVEL                               | 48 |
| 49 |                                                                       | STORE  | W +72D                                 | 49 |
| 50 |                                                                       | STORE  | W +80D                                 | 50 |
| 51 |                                                                       | STORE  | W +88D                                 | 51 |
| 52 |                                                                       | SLOAD  |                                        | 52 |
| 53 |                                                                       |        |                                        | 53 |
| 54 |                                                                       |        |                                        | 54 |
| 55 |                                                                       |        |                                        | 55 |
| 56 |                                                                       |        |                                        | 56 |
| 57 |                                                                       |        |                                        | 57 |
| 58 |                                                                       |        |                                        | 58 |
| 59 |                                                                       |        |                                        | 59 |
| 60 |                                                                       |        |                                        | 60 |

|    |   |
|----|---|
| 76 | 1 |
| 77 |   |

```
1 #
2 # RLC IN PD36D
3 # SIN THETA IN SINTHETA AND PD6
4 # RXZ NORM IN RXZ (N IN X1)
5
6 RADARANG STQ CALL
7 RDRET
8 GETULC
9 VCOMP DOT
10
11 SL1R PUSH # SIN THETA TO PD6
12 STORE SINTHETA
13
14 DSQ BDSU # 1-(SIN THETA)**2
15 SQRT DMP
16
17 SL1 36D
18 NORM X1 # SET SHIFT COUNTER IN X1
19
20 STORE RXZ
21 GOTO # EXIT
22
23 LGCUPDTE STQ CALL
24 LGRET
25 INCORP1
26
27 VLOAD ABVAL
28 LXA,2 DELTAX +6
29
30 STOVL SCALSHFT # 0 -- MOON. 2 -- EARTH
31 2,2 # SET VEL DISPLAY TO B-7
32
33 ABVAL DELTAX
34 SRR*
35 2,2 # SET POS DISPLAY TO B-29
36
37 STORE R22DISP
38 SLOAD SR
39
40 DSU 10D
41 BMN
42 R22DISP
43
44 SLOAD R22LEM96 # GO DISPLAY
45 DSU
46 VMAX
47
48 BMN R22DISP +2 # VMAX MINUS VEL. DEVIATION
49
50 GOTO R22LEM96 # GO DISPLAY
51
52 ASTOK CALL
53 INCORP2
54
55 GOTO
56
57 IMUVAR 2DEC LGRET
58 E-6 B12 # RAD**2
59
60 WSIZE DEC 161
```

2PI/8                      2DEC                      3.141592653 B-2

```
EBANK= LOSCOUNT
```

```

1 # PROGRAM NAME LRS24.1 RR SEARCH ROUTINE
2
3 # MCD NO. 0 BY P. VOLANTE, SDC DATE 1-15-67
4
5 #
6 # FUNCTIONAL DESCRIPTION
7
8 # DRIVES THE RENDEZVOUS RADAR IN A HEXAGONAL SEARCH PATTERN ABOUT THE LOS TO THE CSM (COMPUTED FROM THE CSM AND LM
9 # STATE VECTORS) CHECKING FOR THE DATA GOOD DISCRETE AND MONITORING THE ANGLE BETWEEN THE RADAR BORESIGHT AND THE
10 # LM +Z AXIS. IF THIS ANGLE EXCEEDS 30 DEGREES THE PREFERRED TRACKING ATTITUDE ROUTINE IS CALLED TO PERFORM AN
11 # ATTITUDE MANEUVER.
12 #
13 # CALLING SEQUENCE -- BANKCALL FOR LRS24.1
14 #
15 # SUBROUTINES CALLED
16 #
17 # LEMCONIC R61LEM
18 # CSMCONIC RRDESSM
19 # JOBDELAY FLAGDOWN
20 # WAITLIST FLAGUP
21 # RRNB BANKCALL
22 #
23 # EXIT -- TO ENDOFJOB WHEN THE SEARCH FLAG (SRCHOPT) IS NOT SET
24 #
25 # OUTPUT
26 #
27 # DATAGOOD (SP) -- FOR DISPLAY IN R1 -- 00000 INDICATES NO LOCKON
28 # 11111 INDICATES LOCKON ACHIEVED
29 # OMEGAD (SP) -- FOR DISPLAY IN R2 -- ANGLE BETWEEN RR BORESIGHT VECTOR AND THE SPACECRAFT +Z AXIS
30 #
31 # ERASABLE INITIALIZATION REQUIRED:
32 #
33 # SEARCH FLAG MUST BE SET
34 # LM AND CSM STATE VECTORS AND REFSMMAT MATRIX
35 #
36 # DEBRIS
37 #
38 # RLMSRCH UXVECT
39 # VXRLM UYVECT
40 # LOSDESRD NSRCHPNT
41 # DATAGOOD OMEGAD
42 # MPAC PUSHLIST
43
44 # LRS24.1 COUNT* $$/LRS24
45 # CAF ZERO
46 # CHKSRCH TS NSRCHPNT # SET SEARCH PATTERN POINT COUNTER TO ZERO
47 # CAF BIT14 # ISSUE AUTO TRACK ENABLE TO RADAR
48 # EXTEND
49
50
51
52
53
54
55
56
57
58
59
60

```

LRS24.11

|        |          |                                            |
|--------|----------|--------------------------------------------|
| WOR    | CHAN12   |                                            |
| CAF    | SRCHOBIT | # CHECK IF SEARCH STILL REQUESTED          |
| MASK   | FLAGWRD2 | # (SRCHOPT FLAG SET)                       |
| EXTEND |          |                                            |
| BZF    | ENDOFJOB | # NO-TERMINATE JOB                         |
| CAF    | 6SECONDS | # SCHEDULE TASK TO DRIVE RADAR TO NEXT PT. |
| INHINT |          |                                            |
| TC     | WAITLIST | # IN 6 SECONDS                             |
| EBANK= | LOSCOUNT |                                            |
| 2CADR  | CALLDGCH |                                            |
| RELINT |          |                                            |
| CS     | RADMODES | # IS REMODE IN PROGRESS                    |
| MASK   | REMODBIT |                                            |
| EXTEND |          |                                            |
| BZF    | ENDOFJOB | # YES -- WAIT SIX SECONDS                  |
| TC     | INTPRET  |                                            |
| RTB    | DAD      | # COMPUTE LOS AT PRESENT TIME + 1.5 SEC.   |
|        | LOADTIME |                                            |
|        | 1.5SECS  |                                            |
| STCALL | TDEC1    |                                            |
|        | LEMCONIC | # EXTRAPOLATE LM STATE VECTOR              |
| VLOAD  |          |                                            |
|        | RATT     |                                            |
| STOVL  | RLMSRCH  | # SAVE LEM POSITION                        |
|        | VATT     |                                            |
| STODL  | SAVLEMV  | # SAVE LEM VELOCITY                        |
|        | TAT      |                                            |
| STCALL | TDEC1    | # EXTRAPOLATE CSM STATE VECTOR             |
|        | CSMCONIC | # EXTRAPOLATE CSM STATE VECTOR             |
| VLOAD  | VSU      | # LOS VECTOR = R(CSM) - R(LM)              |
|        | RATT     |                                            |
|        | RLMSRCH  |                                            |
| UNIT   |          |                                            |
| STOVL  | LOSDESRD | # STORE DESIRED LOS                        |
|        | VATT     | # COMPUTE UNIT(V(CM) CROSS R(CM))          |
| UNIT   | VXV      |                                            |
|        | RATT     |                                            |
| UNIT   |          |                                            |
| STORE  | VXRCM    |                                            |
| VLOAD  | VSU      |                                            |
|        | VATT     |                                            |
|        | SAVLEMV  |                                            |
| MXV    | VSL1     | # CONVERT FROM REFERENCE TO STABLE MEMBER  |
|        | REFSMMAT |                                            |
| STORE  | SAVLEMV  | # VLC = V(CSM) - V(LM)                     |
| SLOAD  | BZE      | # CHECK IF N=0                             |

|    |           |            |                                              |                                      |
|----|-----------|------------|----------------------------------------------|--------------------------------------|
| 1  |           |            |                                              |                                      |
| 2  |           | NSRCHPNT   |                                              |                                      |
| 3  |           | DESGLOS    | # YES -- DESIGNATE ALONG LOS                 |                                      |
| 4  | DSU       | BZE        | # IS N=1                                     |                                      |
| 5  |           | ONEOCT     | # YES -- CALCULATE X AND Y AXES OF           |                                      |
| 6  |           | CALCXY     | # SEARCH PATTERN COORDINATE SYSTEM           |                                      |
| 7  | VLOAD     |            | # NO -- ROTATE X-Y AXES TO NEXT SEARCH POINT |                                      |
| 8  |           | UXVECT     |                                              |                                      |
| 9  | STOVL     | UXVECTPR   | # SAVE ORIGINAL X AND Y VECTORS              |                                      |
| 10 |           | UYVECT     | # UXPRIME = ORIGINAL UX                      |                                      |
| 11 | STORE     | UYVECTPR   | # UYPRIME = ORIGINAL UY                      |                                      |
| 12 | VXSC      |            |                                              |                                      |
| 13 |           | SIN60DEG   | # $UX = (\cos 60)UXPR + (\sin 60)UYPR$       |                                      |
| 14 | STOVL     | UXVECT     |                                              |                                      |
| 15 |           | UXVECTPR   |                                              |                                      |
| 16 | VXSC      | VAD        |                                              |                                      |
| 17 |           | COS60DEG   |                                              |                                      |
| 18 |           | UXVECT     |                                              |                                      |
| 19 | UNIT      |            |                                              |                                      |
| 20 | STOVL     | UXVECT     |                                              |                                      |
| 21 |           | UXVECTPR   | # $UY = (-\sin 60)UXPR + (\cos 60)UYPR$      |                                      |
| 22 | VXSC      |            |                                              |                                      |
| 23 |           | SIN60DEG   |                                              |                                      |
| 24 | STOVL     | UYVECT     |                                              |                                      |
| 25 |           | UYVECTPR   |                                              |                                      |
| 26 | VXSC      | VSU        |                                              |                                      |
| 27 |           | COS60DEG   |                                              |                                      |
| 28 |           | UYVECT     |                                              |                                      |
| 29 | UNIT      |            |                                              |                                      |
| 30 | STORE     | UYVECT     |                                              |                                      |
| 31 | OFFCALC   | VXSC       | # OFFSET VECTOR = K(UY)                      |                                      |
| 32 |           | OFFSTFAC   | # LOS VECTOR + OFFSET VECTOR DEFINES         |                                      |
| 33 |           | LOSDESRD   | # DESIRED POINT IN SEARCH PATTERN            |                                      |
| 34 | UNIT      | MXV        |                                              |                                      |
| 35 |           | REFSMMAT   | # CONVERT TO STABLE MEMBER COORDINATES       |                                      |
| 36 | VSL1      |            |                                              |                                      |
| 37 | CONTDMSG  | STOVL      | RRTARGET                                     |                                      |
| 38 |           |            | SAVLEMV                                      |                                      |
| 39 | STORE     |            | LOSVEL                                       |                                      |
| 40 | EXIT      |            |                                              |                                      |
| 41 | INHINT    |            |                                              |                                      |
| 42 | TC        | KILLTASK   | # KILL ANY PRESENTLY WAITLISTED TASK         |                                      |
| 43 | CADR      | DESLOOP +2 | # WHICH WOULD DESIGNATE TO THE LAST          |                                      |
| 44 |           |            | # POINT IN THE PATTERN                       |                                      |
| 45 | CONTDMSG2 | CS         | CDESBIT                                      |                                      |
| 46 |           | MASK       | RADMODES                                     | # SET BIT 15 OF RADMODES TO INDICATE |
| 47 |           | AD         | CDESBIT                                      | # A CONTINUOUS DESIGNATE WANTED.     |
| 48 |           | TS         | RADMODES                                     |                                      |
| 49 |           | TC         | INTPRET                                      |                                      |
| 50 |           |            |                                              |                                      |
| 51 | CALL      |            |                                              |                                      |
| 52 |           |            |                                              |                                      |
| 53 |           |            |                                              |                                      |
| 54 |           |            |                                              |                                      |
| 55 |           |            |                                              |                                      |
| 56 |           |            |                                              |                                      |
| 57 |           |            |                                              |                                      |
| 58 |           |            |                                              |                                      |
| 59 |           |            |                                              |                                      |
| 60 |           |            |                                              |                                      |



|    |          |        |        |                                      |                                          |                                         |    |
|----|----------|--------|--------|--------------------------------------|------------------------------------------|-----------------------------------------|----|
| 1  |          |        |        | 1                                    |                                          |                                         |    |
| 2  | RRDESSM  |        |        | # DESIGNATE RADAR TO RRTARGET VECTOR | 2                                        |                                         |    |
| 3  |          |        |        |                                      | 3                                        |                                         |    |
| 4  | EXIT     |        |        |                                      | 4                                        |                                         |    |
| 5  | TC       |        |        | LIMALARM                             | # LOS NOT IN MODE 2 COVERAGE (P22)       | 5                                       |    |
| 6  | TC       |        |        | LIMALARM                             | # VEHICLE MANEUVER REQUIRED (P20)        | 6                                       |    |
| 7  |          |        |        |                                      |                                          | 7                                       |    |
| 8  |          |        |        |                                      | # COMPUTE OMEGA,ANGLE BETWEEN RR LOS AND | 8                                       |    |
| 9  |          |        |        |                                      | # SPACECRAFT +Z AXIS                     | 9                                       |    |
| 10 | OMEGCALC | EXTEND |        |                                      |                                          | 10                                      |    |
| 11 |          | DCA    | CDUT   |                                      |                                          | 11                                      |    |
| 12 |          | DXCH   | TANGNB |                                      |                                          | 12                                      |    |
| 13 |          |        |        | TC                                   | INTPRET                                  | 13                                      |    |
| 14 |          |        |        | CALL                                 |                                          | 14                                      |    |
| 15 |          |        |        |                                      | RRNB                                     | 15                                      |    |
| 16 |          |        |        | DLOAD                                | ACOS                                     | # OMEGA IS ARCCOSINE OF Z-COMPONENT OF  | 16 |
| 17 |          |        |        |                                      | 36D                                      | # VECTOR COMPUTED BY RRNB (LEFT AT 32D) | 17 |
| 18 |          |        |        | STORE                                | OMEGDISP                                 | # STORE FOR DISPLAY IN R2               | 18 |
| 19 |          |        |        | EXIT                                 |                                          |                                         | 19 |
| 20 |          |        |        | TC                                   | ENDOFJOB                                 |                                         | 20 |
| 21 |          |        |        |                                      |                                          |                                         | 21 |
| 22 |          |        |        |                                      |                                          |                                         | 22 |
| 23 |          |        |        |                                      |                                          |                                         | 23 |
| 24 |          |        |        |                                      |                                          |                                         | 24 |
| 25 |          |        |        |                                      |                                          |                                         | 25 |
| 26 |          |        |        |                                      |                                          |                                         | 26 |
| 27 |          |        |        |                                      |                                          |                                         | 27 |
| 28 |          |        |        |                                      |                                          |                                         | 28 |
| 29 |          |        |        |                                      |                                          |                                         | 29 |
| 30 |          |        |        |                                      |                                          |                                         | 30 |
| 31 |          |        |        |                                      |                                          |                                         | 31 |
| 32 |          |        |        |                                      |                                          |                                         | 32 |
| 33 |          |        |        |                                      |                                          |                                         | 33 |
| 34 |          |        |        |                                      |                                          |                                         | 34 |
| 35 |          |        |        |                                      |                                          |                                         | 35 |
| 36 |          |        |        |                                      |                                          |                                         | 36 |
| 37 |          |        |        |                                      |                                          |                                         | 37 |
| 38 |          |        |        |                                      |                                          |                                         | 38 |
| 39 |          |        |        |                                      |                                          |                                         | 39 |
| 40 |          |        |        |                                      |                                          |                                         | 40 |
| 41 |          |        |        |                                      |                                          |                                         | 41 |
| 42 |          |        |        |                                      |                                          |                                         | 42 |
| 43 |          |        |        |                                      |                                          |                                         | 43 |
| 44 |          |        |        |                                      |                                          |                                         | 44 |
| 45 |          |        |        |                                      |                                          |                                         | 45 |
| 46 |          |        |        |                                      |                                          |                                         | 46 |
| 47 |          |        |        |                                      |                                          |                                         | 47 |
| 48 |          |        |        |                                      |                                          |                                         | 48 |
| 49 |          |        |        |                                      |                                          |                                         | 49 |
| 50 |          |        |        |                                      |                                          |                                         | 50 |
| 51 |          |        |        |                                      |                                          |                                         | 51 |
| 52 |          |        |        |                                      |                                          |                                         | 52 |
| 53 |          |        |        |                                      |                                          |                                         | 53 |
| 54 |          |        |        |                                      |                                          |                                         | 54 |
| 55 |          |        |        |                                      |                                          |                                         | 55 |
| 56 |          |        |        |                                      |                                          |                                         | 56 |
| 57 |          |        |        |                                      |                                          |                                         | 57 |
| 58 |          |        |        |                                      |                                          |                                         | 58 |
| 59 |          |        |        |                                      |                                          |                                         | 59 |
| 60 |          |        |        |                                      |                                          |                                         | 60 |



# CALCULATE X AND Y VECTORS FOR SEARCH PATTERN COORDINATE SYSTEM.

CALCXY

VLOAD

VXV  
VXRCM  
LOSDESRD

UNIT

STOVL

UXVECT  
LOSDESRD

# UX = (VLM X RLM) X LOS

VXV

UNIT  
UXVECT  
UYVECT

# UY = LOS X UX

STORE

GOTO

OFFCALC

DESGLOS

VLOAD

MXV  
LOSDESRD  
REFSMMAT

# WHEN N= 0,DESIGNATE ALONG LOS

# CONVERT LOS FROM REFERENCE TO SM COORDS

VSL1

GOTO  
CONTDESG

CALLDGCH

CAE

MASK  
EXTENDFLAGWRDO  
RNDVZBIT

# IS RENDEZVOUS FLAG SET

BZF

CAF

TC

TASKOVER  
PRIO25  
FINDVAC

# NO -- EXIT R24

# YES -- SCHEDULE JOB TO DRIVE RADAR TO NEXT

# PONT IN SEARCH PATTERN

EBANK=  
2CADRRLMSRCH  
DATGDCHK

TC

TASKOVER

DATGDCHK

CAF

BIT4

# CHECK IF DATA GOOD DISCRETE PRESENT

EXTEND  
RAND  
EXTEND

CHAN33

BZF

CS

AD

STORE1S  
SIX  
NSRCHPNT

# YES -- GO TO STORE 11111 FOR DISPLAY IN R1

# IS N GREATER THAN 6

EXTEND

BZF

INCR

LRS24.1  
NSRCHPNT

# YES -- RESET N = 0 AND START AROUND AGAIN

# NO -- SET N = N+1 AND GO TO

TCF

CHKSRCH

# NEXT POINT IN PATTERN

STORE1S

CAF

TS

ALL1S  
DATAGOOD

# STORE 11111 FOR DISPLAY IN R1

|    |          |        |            |                                                |  |
|----|----------|--------|------------|------------------------------------------------|--|
| 1  |          | INHINT |            |                                                |  |
| 2  |          | TC     | KILLTASK   | # DELETE DESIGNATE TASK FROM                   |  |
| 3  |          | CADR   | DESLOOP +2 | # WAITLIST USING KILLTASK                      |  |
| 4  |          | TC     | ENDOFJOB   |                                                |  |
| 5  | LIMALARM | TC     | ALARM      | # ISSUE ALARM 527 -- LOS NOT IN MODE2          |  |
| 6  |          | OCT    | 527        | # COVERAGE IN P22 OR VEHICLE MANEUVER          |  |
| 7  |          | INHINT |            | # REQUIRED IN P20                              |  |
| 8  |          | TC     | KILLTASK   | # KILL WAITLIST CALL FOR NEXT                  |  |
| 9  |          | CADR   | CALLDGCH   | # POINT IN SEARCH PATTERN                      |  |
| 10 |          | TC     | ENDOFJOB   |                                                |  |
| 11 | ALLIS    | DEC    | 11111      |                                                |  |
| 12 | SIN60DEG | 2DEC   | .86603     |                                                |  |
| 13 | COS60DEG | =      | DPHALF     | # (2DEC .50)                                   |  |
| 14 | UXVECTPR | EQUALS | 12D        | # PREVIOUS                                     |  |
| 15 | UYVECTPR | EQUALS | 18D        |                                                |  |
| 16 | RLMUNIT  | EQUALS | 12D        |                                                |  |
| 17 | OFFSTFAC | 2DEC   | 0.05678    | # TANGENT OF 3.25 DEGREES                      |  |
| 18 | ONEOCT   | OCT    | 00001      | # ***** NOTE -- THESE TWO CONSTANTS MUST ***** |  |
| 19 | 3SECONDS | 2DEC   | 300        | # ***** BE IN THIS ORDER BECAUSE *****         |  |
| 20 |          |        |            | # ***** ONEOCT NEEDS A LOWER ORDER *****       |  |
| 21 |          |        |            | # ***** WORD OF ZEROES *****                   |  |
| 22 | 6SECONDS | DEC    | 600        |                                                |  |
| 23 | 1.5SECS  | 2DEC   | 150        |                                                |  |
| 24 | ZERO/SP  | EQUALS | HI6ZEROS   |                                                |  |
| 25 |          | BLOCK  | 02         |                                                |  |
| 26 |          | SETLOC | FFTAG5     |                                                |  |
| 27 | GOTOV56  | BANK   | \$\$/P20   | # P20 TERMINATES BY GOTOV56 INSTEAD OF         |  |
| 28 |          | COUNT* |            | # GOTOPPOH                                     |  |
| 29 |          | EXTEND |            |                                                |  |
| 30 |          | DCA    | VB56CADR   |                                                |  |
| 31 |          | TCF    | SUPDXCHZ   |                                                |  |
| 32 |          | EBANK= | WHOCARES   |                                                |  |
| 33 | VB56CADR | 2CADR  | TRMTRACK   |                                                |  |
| 34 |          |        |            |                                                |  |
| 35 |          |        |            |                                                |  |
| 36 |          |        |            |                                                |  |
| 37 |          |        |            |                                                |  |
| 38 |          |        |            |                                                |  |
| 39 |          |        |            |                                                |  |
| 40 |          |        |            |                                                |  |
| 41 |          |        |            |                                                |  |
| 42 |          |        |            |                                                |  |
| 43 |          |        |            |                                                |  |
| 44 |          |        |            |                                                |  |
| 45 |          |        |            |                                                |  |
| 46 |          |        |            |                                                |  |
| 47 |          |        |            |                                                |  |
| 48 |          |        |            |                                                |  |
| 49 |          |        |            |                                                |  |
| 50 |          |        |            |                                                |  |
| 51 |          |        |            |                                                |  |
| 52 |          |        |            |                                                |  |
| 53 |          |        |            |                                                |  |
| 54 |          |        |            |                                                |  |
| 55 |          |        |            |                                                |  |
| 56 |          |        |            |                                                |  |
| 57 |          |        |            |                                                |  |
| 58 |          |        |            |                                                |  |
| 59 |          |        |            |                                                |  |
| 60 |          |        |            |                                                |  |

# PROGRAM NAME: R29 (RENDEZVOUS RADAR DESIGNATE DURING POWERED FLIGHT)  
# MOD NO. 2 BY H. BLAIR-SMITH JULY 2, 1968

# FUNCTIONAL DESCRIPTION:

# DESIGNATES THE RENDEZVOUS RADAR TOWARD THE COMPUTES LOS TO THE CSM, WITH THE CHIEF OBJECTIVE OF OBTAINING RANGE AND RANGE RATE DATA AT 2-SECOND INTERVALS FOR TRANSMISSION TO THE GROUND. WHEN THE RR IS WITHIN .5 DEGREE OF THE COMPUTED LOS, TRACKING IS ENABLED, AND DESIGNATION CONTINUES UNTIL THE DATA-GOOD DISCRETE IS RECEIVED. AT THAT POINT, DESIGNATION CEASES AND A RADAR-READING ROUTINE TAKES OVER, PREPARING A CONSISTENT SET OF DATA FOR DOWN TELEMETRY. THE SET INCLUDES RANGE, RANGE RATE, MARK TIME, TWO RR CDU ANGLES, THREE IMUCDU ANGLES, AND AN INDICATOR WHICH IS 1 WHEN THE SET IS CONSISTENT AND 0 OTHERWISE. THE INDICATOR IS IN TRKMKCNT.

# CALLING SEQUENCE: BEGUN EVERY 2 SECONDS AS AN INTEGRAL PART OF SERVICER

# SUBROUTINES CALLED:

|   |          |          |
|---|----------|----------|
| # | REMODE   | RRPONLY  |
| # | UNIT     | MPACVBUF |
| # | QUICTRIG | AX*SR*T  |
| # | SPSIN    | SPCOS    |
| # | SETRRECR | RROUT    |
| # | RRRDOT   | RRRANGE  |

# EXIT: TO NOR29NOW, IN SERVICER.

# OUTPUT: (ALL FOR DOWNLINK)

|   |        |           |                                        |
|---|--------|-----------|----------------------------------------|
| # | RM     | RDOTM     | (RAW)                                  |
| # | AIG    | AMG       |                                        |
| # | AOG    | TRKMKCNT  | TRKMKCNT = 00001 IF SET IS CONSISTENT, |
| # | TANGNB | TANGNB +1 | OTHERWISE TRKMKCNT = 00000.            |
| # | MKTIME |           |                                        |

```
#
ERASABLE INITIALIZATION REQUIRED:
#
NOR29FLG READRFLG (TO 1 AND 0 BY FRESH START) (RESET NOR29FLG TO LET SERVICER RUN R29)
PIPTIME RADMODES (BIT 10) (BIT SET TO 0 BY FRESH START)
R(CSM) V(CSM)
R V (PIPTIME THRU V BY AVE G IN SERVICER)
#
DEBRIS:
#
RADMODES (BIT 10)
LOSSM LOSVDT/4 (= RRTARGET & LOSVEL)
SAVECDUT OLDESFLG (SAVECDUT = MLOSV)
LOSCMFLG READRFLG
#
ALARMS: NONE.
#
COMPONENT JOBS AND TASKS:
#
INITIALIZING, IF RR IS FOUND TO BE IN MODE 1: JOB R29REMOJ AND TASK REMODE: ALWAYS: TASK PREPOS29.
DESIGNATING: TASK BEGDES29 & JOB R29DODES.
RADAR READING: TASK R29READ AND JOB R29RDJOB. ALL JOBS ARE NOVAC TYPE.
#
BANK 33
SETLOC R29/SERV
BANK
#
COUNT* $$/R29
#
NR29&RDR EQUALS EBANK5
```

# SERVICER COMES TO R29 FROM "R29?" IF NOR29FLG, READRFLG, RRREMODE, RRCDUZRO, RRREPOS, AND DISPLAY-INERTIAL-DATA  
# ARE ALL RESET, AND THE RR IS IN LGC MODE (OFTEN CONFUSINGLY CALLED AUTO MODE).

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| R29      | CS     | RADMODES |                                            |
|          | MASK   | DESIGBIT |                                            |
|          | EXTEND |          |                                            |
|          | BZF    | R29.LOS  | # BRANCH IF DESIGNATION IS ALREADY ON.     |
|          | INHINT |          |                                            |
|          | ADS    | RADMODES | # SHOW THAT DESIGNATION IS NOW ON.         |
|          | CS     | BIT14    |                                            |
|          | EXTEND |          |                                            |
|          | WAND   | CHAN12   | # REMOVE RR TRACK ENABLE DISCRETE.         |
|          | CS     | LOSCMBIT |                                            |
|          | MASK   | FLAGWRD2 |                                            |
|          | TS     | FLAGWRD2 | # CLEAR LOSCMFLG TO SHOW DES. LOOP IS OFF. |
|          | CS     | OLDESBIT |                                            |
|          | MASK   | STATE    |                                            |
|          | TS     | STATE    | # SHOW THAT DES. LOOP IS NOT REQUESTED.    |
|          | TC     | BANKCALL |                                            |
|          | CADR   | SETRRECR | # ENABLE RR ERROR COUNTERS.                |
|          | CA     | ANTENBIT |                                            |
|          | MASK   | RADMODES |                                            |
|          | CCS    | A        | # TEST RR MODE BIT.                        |
|          | TCF    | SETPRPOS | # MODE 2.                                  |
|          | CA     | PRI021   | # MODE 1: MUST REMODE.                     |
|          | TC     | NOVAC    |                                            |
|          | EBANK= | LOSCOUNT |                                            |
|          | 2CADR  | R29REMOJ | # NEEDS OWN JOB TO RADSTALL IN.            |
|          | CS     | DESIGBIT |                                            |
|          | MASK   | RADMODES | # CLEAR DESIGNATE FLAG IN RADMODES         |
|          | TS     | RADMODES | # BEFORE CALLING REMODE                    |
|          | CA     | REMODBIT |                                            |
|          | ADS    | RADMODES | # SHOW THAT REMODING IS ON.                |
|          | TCF    | NOR29NOW | # CONTINUE SERVICER FUNCTIONS.             |
| SETPRPOS | CA     | ONE      |                                            |
|          | TC     | WAITLIST |                                            |
|          | EBANK= | LOSCOUNT |                                            |
|          | 2CADR  | PREPOS29 | # TASK TO SET TRUNNION ANGLE TO 180 DEG.   |
|          | CA     | REPOSBIT |                                            |
|          | ADS    | RADMODES | # SHOW THAT REPOSITIONING IS ON.           |
|          | TCF    | NOR29NOW |                                            |

# FORCE RENDEZVOUS RADAR INTO MODE 2.

|          |        |          |                              |
|----------|--------|----------|------------------------------|
| R29REMOJ | CA     | ONE      |                              |
|          | TC     | WAITLIST |                              |
|          | EBANK= | LOSCOUNT |                              |
|          | 2CADR  | REMODE   | # REMODE MUST RUN AS A TASK. |
|          | TC     | BANKCALL | # WAIT FOR END OF REMODING   |
|          | CADR   | RADSTALL |                              |
|          | TCF    | ENDOFJOB | # BAD EXIT CAN'T HAPPEN.     |
|          | TCF    | ENDOFJOB |                              |

# TASK TO PREPOSITION THE RR TRUNNION ANGLE TO -180 DEG.

SETLOC R29S1  
BANK

|          |      |          |                                   |
|----------|------|----------|-----------------------------------|
| PREPOS29 | CA   | NEGMAX   | # -180 DEG.                       |
|          | TC   | RRONLY   | # DRIVE TRUNNION CDU.             |
|          | CS   | REPOSBIT | # SHOW THAT REPOSITIONING IS OFF. |
|          | MASK | RADMODES |                                   |
|          | TS   | RADMODES |                                   |
|          | TCF  | TASKOVER |                                   |

# COMPUTE THE LINE-OF-SIGHT AND LOS VELOCITY, AND PASS THEM TO THE R29DODES LOOP.

SETLOC R29  
BANK

|         |        |         |                                    |
|---------|--------|---------|------------------------------------|
| R29.LOS | EXTEND |         |                                    |
|         | DCS    | PIPTIME |                                    |
|         | DXCH   | MPAC    |                                    |
|         | EXTEND |         |                                    |
|         | DCA    | TIME2   |                                    |
|         | DAS    | MPAC    | # (MPAC) = T-PIPTIME, SCALED B-28. |
|         | TS     | MODE    | # SET MODE TO DOUBLE PRECISION.    |
|         | CA     | MPAC +1 |                                    |
|         | EXTEND |         |                                    |
|         | MP     | BIT12   |                                    |
|         | DXCH   | MPAC    | # T-PIPTIME NOW SCALED B-17.       |
|         | TC     | INTPRET |                                    |

```
LOSCMFLG = 0 MEANS THAT THE DESIGNATION IS READY FOR NEW DATA. SETTING LOSCMFLG MAKES IT GO AWAY SO SETUP29D CAN
START IT UP WHEN THE DATA IS IN PLACE.
```

```
PDVL VSU # PUSH DOWN T-PIPTIME
```

```
V(CSM)
```

```
V # LOSVEL = V(CSM) - V
```

```
PDDL VXSC # SWAP LOSVEL FOR T-PIPTIME, MULTIPLY THEM
```

```
VAD VSU # AND ADD THE RESULT TO R(CSM) - R TO GET
```

```
R(CSM) # AN UP-TO-DATE LOS VECTOR IN SM AXES.
```

```
R
```

```
BOFSET EXIT # (BOFSET DOES ITS THING INHINTED.)
```

```
LOSCMFLG # IF DESIGNATE LOOP IS OFF, CHANGE LOSCM-
```

```
SETUP29D # FLG TO ON AND GO TO SET UP NEW DATA.
```

```
TCF NOR29NOW # IF DES. LOOP IS ON, LET IT USE OLD DATA.
```

```
SETUP29D STOVL LOSSM # LINE-OF-SIGHT VECTOR, STABLE MEMBER AXES
```

```
0
```

```
VXSC
```

```
.5SECB17
```

```
STORE LOSVDT/4 # 1/2 SECOND'S WORTH OF LOS VELOCITY.
```

```
CLEAR EXIT
```

```
LOSCMFLG # LET R29DLOOP USE NEW DATA.
```

```
CS STATE
```

```
MASK OLDESBIT
```

```
EXTEND
```

```
BZF NOR29NOW # BRANCH IF R29 DES. LOOP IS REQUESTED.
```

```
INHINT
```

```
ADS STATE # OTHERWISE REQUEST IT NOW.
```

```
CCS PIPCTR # SEE IF TASK SHOULD BE OFFSET ONE SECOND.
```

```
CS SUPER110 # -96D +100D = 4.
```

```
AD 1SEC # 0 +100D = 100D.
```

```
TC WAITLIST
```

```
EBANK= LOSCOUNT
```

```
2CADR BEGDES29 # START BEGDES29 TASK ASAP.
```

```
TCF NOR29NOW # RELINT AND CONTINUE SERVICER FUNCTIONS.
```

```
.5SECB17 2DEC 50 B-17
```

# R29 DESIGNATE JOB AND TASK MACHINERY. TASK RECURS EVERY .5 SEC UNTIL DESIGNATE IS CALLED OFF; IT MAY WAIT FOR A  
# CENTISECOND OR TWO IF IT COMES UP WHILE SETUP29D IS SUPPLYING NEW DATA.

BANK 24  
SETLOC P20S

BANK

COUNT\* \$\$/R29

BEGDES29

CAF PRI021  
TC NOVAC

EBANK= LOSVDT/4  
2CADR R29DODES

# START R29DODES JOB TWICE A SECOND.

R29DLOOP

CAF .5SEC  
TC VARDELAY

CS RADMODES  
MASK DESIGBIT  
CCS A

TCF TASKOVER

# QUIT IF DESIGNATION IS CALLED OFF.

CS FLAGWRD2  
MASK LOSCMBIT

EXTEND  
BZF

+3

# BRANCH IF SETUP29D'S SUPPLYING NEW DATA.

ADS FLAGWRD2  
TCF BEGDES29

# SET LOSCMFLG: SHOW THAT DES. LOOP IS ON.

CA ONE  
TCF R29DLOOP +1

# WAIT A CENTISECOND FOR NEW DATA.



# R29DODES: RR DESIGNATION LOOP FOR R29  
#

# THIS ROUTINE DOES MUCH THE SAME THING AS DODES, BUT A GREAT DEAL FASTER. IT TAKES THE NON-UNITIZED LOS VECTOR  
# IN STABLE MEMBER COORDINATES (LOSSM) AND A DELTA-LOS IN SM AXES (LOSVD/4) WHICH IS 1/2 SEC TIMES LOS VELOCITY,  
# AND DEVELOPS THE SHAFT AND TRUNNION COMMANDS USING SINGLE PRECISION AS MUCH AS POSSIBLE, AND INTERPRETIVE NOT AT  
# ALL. THE UNIT(LOSM + LOSVEL \* 1 SEC) IS COMPUTED IN DP AND TRANSFORMED TO NAV BASE COORDINATES IN DOUBLE PRE-  
# CISION (USING SP SINES AND COSINES OF CDU ANGLES), AND THE REST IS DONE IN SP.

# THE FUNCTIONAL DIFFERENCE IS THAT R29DODES ALWAYS CLEARS LOSCMFLG WHEN IT ENDS, AND IT STARTS UP THE R29READ  
# TASK WHEN LOCK-ON IS ACHIEVED.

BANK 32  
SETLOC F2DPS\*32  
BANK

COUNT\* \$\$/R29  
EBANK= LOSVD/4

R29DODES CA ONE  
TS TANG # INDICATE 1ST PASS THRU VECTOR LOOP.  
CA FIVE

R29DVBEG CCS A # COUNT DOWN BY TWO'S IN VECTOR LOOP.

TS Q  
CCS TANG  
TCF R29DPAS1 # DO THIS ON 1ST PASS THRU LOOP.

EXTEND # (A "PASS" HERE MEANS 3 TIMES AROUND).  
INDEX Q

DCA LOSVD/4  
INDEX Q  
DAS LOSSM # ADVANCE LOS VECTOR 1/2 SECOND.

R29DPAS1 EXTEND  
INDEX Q  
DCA LOSSM  
INDEX Q # MOVE CURRENT LOS (1ST PASS) OR LOS PRO-  
DXCH MPAC +1 # JECTED 1/2 SEC AHEAD (2ND PASS).

CCS TANG  
TCF R29DVEND # BUG OUT HERE IN 1ST PASS.

EXTEND  
INDEX Q  
DCA LOSVD/4  
INDEX Q  
DAS MPAC +1 # PROJECT LOS 1 SECOND AHEAD (2ND PASS).

R29DVEND CCS Q  
TCF R29DVBEG # BRANCH TO CONTINUE VECTOR LOOP.

# UNITIZE AND TRANSFORM TO NAV BASE AXES THE PRESENT LOS (1ST PASS) OR THE 1-SEC PROJECTED LOS (2ND PASS).

|          |            |                                           |
|----------|------------|-------------------------------------------|
| DXCH     | MPAC +1    |                                           |
| DXCH     | MPAC       |                                           |
| CA       | R29FXLOC   | # = ADRES INTB15 + -34D                   |
| TS       | FIXLOC     |                                           |
| TC       | USPRCADR   | # WITH FIXLOC ARMED FOR LENGTH AND LENGTH |
| CADR     | UNIT       | # SQUARED, BORROW UNITIZING ROUTINE.      |
| TC       | MPACVBUF   | # MOVE UNIT(LOS) TO AX*SR*T ARG AREA.     |
| CCS      | TANG       |                                           |
| TCF      | +2         |                                           |
| TCF      | GOTANGLS   | # GET CDU ANGLES ONLY AFTER 1ST PASS.     |
| INHINT   |            | # ENSURE CONSISTENT CDU READINGS.         |
| EXTEND   |            |                                           |
| DCA      | CDUT       |                                           |
| DXCH     | SAVECDUT   | # TRUNNION AND SHAFT ANGLES.              |
| CA       | CDUY       |                                           |
| TS       | CDUSPOT    |                                           |
| CA       | CDUZ       |                                           |
| TS       | CDUSPOT +2 |                                           |
| CA       | CDUX       |                                           |
| TS       | CDUSPOT +4 | # CDU ANGLES IN FUNNY ORDER FOR AX*SR*T.  |
| TC       | BANKCALL   |                                           |
| CADR     | QUICTRIG   | # GET SINES AND COSINES OF CDU ANGLES.    |
| GOTANGLS | CS         | THREE                                     |
|          | TC         | BANKCALL                                  |
|          | CADR       | AX*SR*T                                   |
|          |            | # TRANSFORM UNIT LOS TO NB AXES (ULOSNB). |
|          | CCS        | TANG                                      |
|          | TCF        | +2                                        |
|          | TCF        | R29DPAS2                                  |
|          |            | # GO TO RR COMMAND COMP. AFTER 2ND PASS.  |

# COMPUTE COSINE OF THE ANGLE BETWEEN THE PRESENT LOS AND THE RR BORESIGHT VECTOR, AND SET THE SELFTRACK ENABLE IF  
# THE COSINE IS APPROXIMATELY COS(.5 DEG) OR GREATER (I.E., SMALLER ANGLE).

INHINT

TS TANG # INDICATE 2ND PASS THRU VECTOR LOOP.

CA SAVECDUT

TC SPCOS

TS PUSHLOC # PUSHLOC = COS T.

CS SAVECDUT

TC SPSIN

TS MODE # MODE = -SIN T.

EXTEND

MP VBUF +2 # FORM - SIN T ULOSBNBY.

DXCH MPAC

CA SAVECDUT +1

TC SPSIN

TS SAVECDUT # SAVECDUT NOW = SIN S.

EXTEND

MP PUSHLOC

EXTEND

MP VBUF

DAS MPAC

CA SAVECDUT +1

TC SPCOS

TS SAVECDUT +1 # SAVECDUT +1 NOW = COS S.

EXTEND

MP PUSHLOC

EXTEND

MP VBUF +4

DAS MPAC

EXTEND

DCA MPAC

TESTCOS

DAS MPAC

CCS A

CA BIT14

NOOP

EXTEND

WOR CHAN12

RELINT

TCF

R29DVBEG -1 # MAKE 2ND PASS THRU VECTOR LOOP.

# COMPUTE SHAFT AND TRUNNION COMMANDS TO NULL HAVE THE ERROR IN HALF A SECOND.

|          |        |             |                                        |
|----------|--------|-------------|----------------------------------------|
| R29DPAS2 | CA     | SAVECDUT +1 |                                        |
|          | EXTEND |             |                                        |
|          | MP     | VBUF        | # FORM COS S ULOSNB'X.                 |
|          | DXCH   | TANG        |                                        |
|          | CS     | SAVECDUT    |                                        |
|          | EXTEND |             |                                        |
|          | MP     | VBUF +4     | # FORM - SIN S ULOSNB'Z.               |
|          | DAS    | TANG        | # RAW SHAFT CMD = ULOSNB' . (COS S, 0, |
|          | CS     | MODE        | # - SIN S)                             |
|          | EXTEND |             |                                        |
|          | MP     | SAVECDUT    |                                        |
|          | EXTEND |             |                                        |
|          | MP     | VBUF        | # FORM SIN T SIN S ULOSNB'X.           |
|          | DXCH   | MPAC        |                                        |
|          | CA     | PUSHLOC     |                                        |
|          | EXTEND |             |                                        |
|          | MP     | VBUF +2     | # FORM COS T ULOSNB'Y.                 |
|          | DAS    | MPAC        |                                        |
|          | CS     | MODE        |                                        |
|          | EXTEND |             |                                        |
|          | MP     | SAVECDUT +1 |                                        |
|          | EXTEND |             |                                        |
|          | MP     | VBUF +4     | # FORM SIN T COS S ULOSNB'Z.           |
|          | DAS    | MPAC        | # RAW TRUNNION CMD = ULOSNB'.          |
|          | CA     | MPAC        | # (SIN S SIN T, COS T, SIN S COS T).   |
|          | EXTEND |             |                                        |
|          | MP     | RR29GAIN    |                                        |
|          | XCH    | TANG        | # STORE REFINED T CMD, GET RAW S CMD.  |
|          | EXTEND |             |                                        |
|          | MP     | RR29GAIN    |                                        |
|          | TS     | TANG +1     | # STORE REFINED S CMD.                 |

# WHETHER OR NOT TRACKING WAS ENABLED THIS TIME, CHECK ON RR DATA-GOOD. IF PRESENT, STOP DESIGNATING AND START  
# READING DATA FROM THE RENDEZVOUS RADAR.

|          |                          |                                              |                                                                                  |
|----------|--------------------------|----------------------------------------------|----------------------------------------------------------------------------------|
| DGOOD?   | CAF<br>EXTEND            | BIT4                                         |                                                                                  |
|          | RAND<br>INHINT<br>EXTEND | CHAN33                                       | # GET RR DATA-GOOD BIT.<br># (MAINLY FOR RRROUT).                                |
|          | BZF                      | R29LOKON                                     | # BRANCH IF DATA-GOOD IS PRESENT.                                                |
|          | TC<br>CADR<br>TCF        | BANKCALL<br>RRROUT<br>END29DOD               | # DATA-GOOD IS ABSENT, SO SEND COMMANDS.                                         |
| R29LOKON | CS<br>MASK<br>TS         | DESIGBIT<br>RADMODES<br>RADMODES             | # SHOW THAT DESIGNATION IS OVER.                                                 |
|          | CS<br>EXTEND             | BIT2                                         |                                                                                  |
|          | WAND                     | CHAN12                                       | # DISABLE RR ERROR COUNTERS.                                                     |
|          | CA<br>ADS<br>CCS         | READRBIT<br>FLAGWRD3<br>PIPCTR               | # SHOW THAT READING HAS BEEN REQUESTED.<br># SEE IF TASK SHOULD BE OFFSET 1 SEC. |
|          | CS<br>AD<br>TC           | SUPER110<br>1SEC<br>WAITLIST                 | # -96D + 100D = 4.<br># 0 + 100D = 100D.                                         |
|          | EBANK=<br>2CADR          | LOSCOUNT<br>R29READ                          | # START READING TASK AND JOB.                                                    |
| END29DOD | CS<br>MASK<br>TS<br>TCF  | LOSCMBIT<br>FLAGWRD2<br>FLAGWRD2<br>ENDOFJOB | # ALWAYS CLEAR LOSCMFLG.                                                         |
| R29FXLOC | ADRES                    | INTB15+ -34D                                 |                                                                                  |
| RR29GAIN | DEC                      | -.53624                                      |                                                                                  |
| LOSVDT/4 | EQUALS                   | LOSVEL                                       |                                                                                  |
| LOSSM    | EQUALS                   | RRTARGET                                     |                                                                                  |
| SAVECDUT | EQUALS                   | MLOSV                                        |                                                                                  |

# RR READING IS SET UP BY R29DODES WHEN IT DETECTS RR LOCK-ON

BANK 24  
SETLOC P20S  
BANK

COUNT\* \$\$/R29

EBANK= LOSCOUNT

R29READ CAF PRI026 # CALLED BY WAITLIST

TC NOVAC  
EBANK= LOSCOUNT  
2CADR R29RDJOB # START JOB TO READ AND DOWNLINK FOR R29.

CA 2SECS  
TC VARDELAY

CA FLAGWRD3 # 2 SECONDS LATER, SEE IF READING IS STILL  
MASK READRBIT # ALLOWED (NO TRACKER FAIL ETC.)

CCS A  
TCF R29READ # IT'S OK: CALL IT AGAIN.  
TCF TASKOVER # IT AIN'T: WAIT FOR REDESIGNATE.

R29RDJOB CA FLAGWRD3 # CALLED VIA NOVAC.

MASK NR29FBIT  
CCS A # TEST "NOR29FLG".  
TCF ENDRRD29 # R29 IS NOW OVER, STOP AT ONCE.

CA RADMODES  
MASK AUTOMBIT  
CCS A # TEST RR-NOT-IN-AUTO-MODE BIT.  
TCF ENDRRD29 # ASTRO TOOK RR OUT OF AUTO MODE.

TC BANKCALL  
CADR RRRDOT # INITIATE READING OF RANGE RATE.  
TC BANKCALL  
CADR RADSTALL # GOT TO SLEEP UNTIL IT'S READY.  
TCF ENDRRD29 # BAD READ; REDESIGNATE.

# R29 RADAR READING CONTINUED.

DXCH TIMEHOLD  
DXCH MPAC # TIME OF RR READING, FOR DOWNLINK.  
INHINT # BE SURE OF 5 CONSISTENT CDU ANGLES.

EXTEND  
DCA CDUT  
DXCH MPAC +2 # RRCU ANGLES AT RR READ, FOR DOWNLINK.

EXTEND  
DCA CDUY  
DXCH MPAC +4 # MPAC'S 7 WORDS ARE BUFFERED FOR COPYCYCLE.

CA CDUX  
TS MPAC +6 # IMUCDU ANGLES AT RR READ, FOR DOWNLINK.

R29RANGE TC BANKCALL  
CADR RRRANGE # INITIATE READING OF RR RANGE.  
TC BANKCALL

CADR RADSTALL # GO TO SLEEP UNTIL IT'S READY.  
TCF R29RRR? # BAD READ OR SCALE CHANGE ... WHICH?

INHINT  
DXCH DNRRANGE # COPYCYCLE TO LAY OUT NEW R29 DOWNLINK.  
DXCH RM

DXCH MPAC  
DXCH MKTIME  
DXCH MPAC +2

DXCH TANGNB  
DXCH MPAC +4  
DXCH AIG

CA MPAC +6  
TS AOG  
CA ONE

TS TRKMKCNT # SHOW THAT DOWNLINK DATA IS CONSISTENT.  
TCF ENDOFJOB

R29RRR? CS FLAGWRD5  
MASK BIT10  
CCS A # WAS IT A SCALE CHANGE (REAL OR PHONY)?  
TCF ENDRRD29 # NO, A BAD READ; REDESIGNATE.

TC DOWNFLAG  
ADRES RNGSCFLG  
TCF R29RANGE # YES; CLEAR FLAG AND READ AGAIN.

ENDRRD29 CA ZERO # TROUBLE MADE US COME HERE TO LEAVE THE  
TS TRKMKCNT # RR-READING MODE. DISCREDIT DOWNTEL.

TC DOWNFLAG  
ADRES READRFLG  
CS BIT14  
EXTEND



- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

WAND  
TCF

CHAN12  
ENDOFJOB

# REMOVE TRACK-ENABLE DISCRETE.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65
- 66
- 67
- 68
- 69
- 70
- 71
- 72
- 73
- 74
- 75
- 76
- 77
- 78
- 79
- 80



# W-MATRIX MONITOR

BANK 31  
 SETLOC VB67  
 BANK  
 COUNT\* \$\$/EXTVB  
 EBANK= WWPOS

V67CALL TC INTPRET  
 CALL

V67WW

EXIT  
 EXTEND # SAVE THE PRESENT N99 VALUES FOR  
 # COMPARISON AFTER THE DISPLAY

DCA WWPOS  
 DXCH WWBIAS +2  
 EXTEND

DCA WWVEL  
 DXCH WWBIAS +4  
 EXTEND

V06N99DS DCA WWBIAS  
 DXCH WWBIAS +6  
 CAF V06N99

TC BANKCALL  
 CADR GOXDSPF  
 TCF ENDEXT

V6N99PRO TCF V6N99PRO  
 TCF V06N99DS  
 ZL

N99LOOP CA FIVE  
 TS Q  
 INDEX Q

CS WWPOS  
 INDEX Q  
 AD WWPOS +6

ADS L  
 CCS Q # THE SUM OF ALL DIFFERENCES MUST BE ZERO.  
 TCF N99LOOP

LXCH A  
 EXTEND  
 BZF V06N9933

TC UPFLAG  
 ADRES V67FLAG

V06N9933 TC INTPRET  
 BON EXIT  
 V67FLAG

+2  
 TCF ENDEXT  
 DLOAD

|    |          |        |                                    |                          |
|----|----------|--------|------------------------------------|--------------------------|
| 1  |          |        | WWPOS                              |                          |
| 2  |          |        | SL1                                |                          |
| 3  |          | SL4    | SL1                                |                          |
| 4  |          | STODL  | 0D                                 |                          |
| 5  |          |        | WWVEL                              |                          |
| 6  |          | STODL  | 2D                                 |                          |
| 7  |          |        | WWBIAS                             |                          |
| 8  |          | SL     | # SHIFT FROM NOUN SCALING (B-5) TO |                          |
| 9  |          |        | 10D                                | # INTERNAL SCALING (B+5) |
| 10 |          | STORE  | 4D                                 |                          |
| 11 |          | BON    | LXA,1                              |                          |
| 12 |          |        | SURFFLAG                           |                          |
| 13 |          |        | V67SURF                            |                          |
| 14 |          |        | 0D                                 |                          |
| 15 |          | SXA,1  | LXA,1                              |                          |
| 16 |          |        | WRENDPOS                           |                          |
| 17 |          |        | 2D                                 |                          |
| 18 |          | SXA,1  | GOTO                               |                          |
| 19 |          |        | WRENDVEL                           |                          |
| 20 |          |        | V67CLRF                            |                          |
| 21 | V67SURF  | LXA,1  | SXA,1                              |                          |
| 22 |          |        | 0D                                 |                          |
| 23 |          |        | WSURFPOS                           |                          |
| 24 |          | LXA,1  | SXA,1                              |                          |
| 25 |          |        | 2D                                 |                          |
| 26 |          |        | WSURFVEL                           |                          |
| 27 | V67CLRF  | LXA,1  | SXA,1                              |                          |
| 28 |          |        | 4D                                 |                          |
| 29 |          |        | WTRUN                              |                          |
| 30 |          | SXA,1  |                                    |                          |
| 31 |          |        | WSHAFT                             |                          |
| 32 |          | CLEAR  | EXIT                               |                          |
| 33 |          |        | RENDWFLG                           |                          |
| 34 |          | TCF    | ENDEXT                             |                          |
| 35 | V67WW    | STQ    | BOV                                |                          |
| 36 |          |        | S2                                 |                          |
| 37 |          |        | +1                                 |                          |
| 38 |          | CLEAR  | CALL                               |                          |
| 39 |          |        | V67FLAG                            |                          |
| 40 |          |        | INTSTALL                           |                          |
| 41 |          | SSP    | DLOAD                              |                          |
| 42 |          |        | S1                                 |                          |
| 43 |          | DEC    | 6                                  |                          |
| 44 |          |        | ZEROVECS                           |                          |
| 45 |          | STORE  | WWPOS                              |                          |
| 46 |          | STORE  | WWVEL                              |                          |
| 47 |          | STORE  | WWBIAS                             |                          |
| 48 |          | AXT,1  |                                    |                          |
| 49 |          | DEC    | 54                                 |                          |
| 50 | NXPOSVEL | VLOAD* | VSQ                                |                          |
| 51 |          |        | W +54D,1                           |                          |
| 52 |          |        |                                    |                          |
| 53 |          |        |                                    |                          |
| 54 |          |        |                                    |                          |
| 55 |          |        |                                    |                          |
| 56 |          |        |                                    |                          |
| 57 |          |        |                                    |                          |
| 58 |          |        |                                    |                          |
| 59 |          |        |                                    |                          |
| 60 |          |        |                                    |                          |

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

|    |    |
|----|----|
| 1  | 2  |
| 3  | 4  |
| 5  | 6  |
| 7  | 8  |
| 9  | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |
| 35 | 36 |
| 37 | 38 |
| 39 | 40 |
| 41 | 42 |
| 43 | 44 |
| 45 | 46 |
| 47 | 48 |
| 49 | 50 |
| 51 | 52 |
| 53 | 54 |
| 55 | 56 |
| 57 | 58 |
| 59 | 60 |
| 61 | 62 |
| 63 | 64 |
| 65 | 66 |
| 67 | 68 |
| 69 | 70 |
| 71 | 72 |
| 73 | 74 |
| 75 | 76 |
| 77 | 78 |
| 79 | 80 |

1



|    |         |      |            |    |
|----|---------|------|------------|----|
| 1  |         |      |            | 1  |
| 2  |         |      |            | 2  |
| 3  |         | EXIT |            | 3  |
| 4  |         | TC   | POSTJUMP   | 4  |
| 5  |         | CADR | INTWAKE    | 5  |
| 6  | FT99999 | 2DEC | 30479 B-19 | 6  |
| 7  |         |      |            | 7  |
| 8  |         |      |            | 8  |
| 9  |         |      |            | 9  |
| 10 |         |      |            | 10 |
| 11 |         |      |            | 11 |
| 12 |         |      |            | 12 |
| 13 |         |      |            | 13 |
| 14 |         |      |            | 14 |
| 15 |         |      |            | 15 |
| 16 |         |      |            | 16 |
| 17 |         |      |            | 17 |
| 18 |         |      |            | 18 |
| 19 |         |      |            | 19 |
| 20 |         |      |            | 20 |
| 21 |         |      |            | 21 |
| 22 |         |      |            | 22 |
| 23 |         |      |            | 23 |
| 24 |         |      |            | 24 |
| 25 |         |      |            | 25 |
| 26 |         |      |            | 26 |
| 27 |         |      |            | 27 |
| 28 |         |      |            | 28 |
| 29 |         |      |            | 29 |
| 30 |         |      |            | 30 |
| 31 |         |      |            | 31 |
| 32 |         |      |            | 32 |
| 33 |         |      |            | 33 |
| 34 |         |      |            | 34 |
| 35 |         |      |            | 35 |
| 36 |         |      |            | 36 |
| 37 |         |      |            | 37 |
| 38 |         |      |            | 38 |
| 39 |         |      |            | 39 |
| 40 |         |      |            | 40 |
| 41 |         |      |            | 41 |
| 42 |         |      |            | 42 |
| 43 |         |      |            | 43 |
| 44 |         |      |            | 44 |
| 45 |         |      |            | 45 |
| 46 |         |      |            | 46 |
| 47 |         |      |            | 47 |
| 48 |         |      |            | 48 |
| 49 |         |      |            | 49 |
| 50 |         |      |            | 50 |
| 51 |         |      |            | 51 |
| 52 |         |      |            | 52 |
| 53 |         |      |            | 53 |
| 54 |         |      |            | 54 |
| 55 |         |      |            | 55 |
| 56 |         |      |            | 56 |
| 57 |         |      |            | 57 |
| 58 |         |      |            | 58 |
| 59 |         |      |            | 59 |
| 60 |         |      |            | 60 |





|    |                               |        |           |    |
|----|-------------------------------|--------|-----------|----|
| 1  |                               |        |           | 1  |
| 2  | LITIT                         | EXTEND |           | 2  |
| 3  |                               | QXCH   | ITEMP6    | 3  |
| 4  |                               | TC     | TRKFLO +1 | 4  |
| 5  |                               |        |           | 5  |
| 6  |                               | EXTEND |           | 6  |
| 7  |                               | QXCH   | ITEMP6    | 7  |
| 8  |                               | TCF    | RRTRKF    | 8  |
| 9  |                               |        |           | 9  |
| 10 |                               |        |           | 10 |
| 11 | ONLITES                       | INDEX  | ITEMP5    | 11 |
| 12 |                               | CS     | HFLSHBIT  | 12 |
| 13 |                               | MASK   | FLGWRD11  | 13 |
| 14 |                               | TS     | FLGWRD11  | 14 |
| 15 |                               |        |           | 15 |
| 16 |                               | CA     | L         | 16 |
| 17 |                               | TCF    | LITIT     | 17 |
| 18 | VLIGHT                        | TS     | ITEMP5    | 18 |
| 19 |                               | CA     | VLITE     | 19 |
| 20 |                               | TS     | L         | 20 |
| 21 |                               | CA     | BIT8      | 21 |
| 22 |                               | TCF    | BOTHLITS  | 22 |
| 23 |                               |        |           | 23 |
| 24 | HLITE                         | EQUALS | BIT5      | 24 |
| 25 | VLITE                         | EQUALS | BIT3      | 25 |
| 26 |                               |        |           | 26 |
| 27 | # *** END OF LEMP20S .127 *** |        |           | 27 |
| 28 |                               |        |           | 28 |
| 29 |                               |        |           | 29 |
| 30 |                               |        |           | 30 |
| 31 |                               |        |           | 31 |
| 32 |                               |        |           | 32 |
| 33 |                               |        |           | 33 |
| 34 |                               |        |           | 34 |
| 35 |                               |        |           | 35 |
| 36 |                               |        |           | 36 |
| 37 |                               |        |           | 37 |
| 38 |                               |        |           | 38 |
| 39 |                               |        |           | 39 |
| 40 |                               |        |           | 40 |
| 41 |                               |        |           | 41 |
| 42 |                               |        |           | 42 |
| 43 |                               |        |           | 43 |
| 44 |                               |        |           | 44 |
| 45 |                               |        |           | 45 |
| 46 |                               |        |           | 46 |
| 47 |                               |        |           | 47 |
| 48 |                               |        |           | 48 |
| 49 |                               |        |           | 49 |
| 50 |                               |        |           | 50 |
| 51 |                               |        |           | 51 |
| 52 |                               |        |           | 52 |
| 53 |                               |        |           | 53 |
| 54 |                               |        |           | 54 |
| 55 |                               |        |           | 55 |
| 56 |                               |        |           | 56 |
| 57 |                               |        |           | 57 |
| 58 |                               |        |           | 58 |
| 59 |                               |        |           | 59 |
| 60 |                               |        |           | 60 |

```
1 # PROGRAM DESCRIPTION P30 DATE 3-6-67
2 #
3
4 # MOD.1 BY RAMA AIYAWAR
5 # FUNCTIONAL DESCRIPTION
6 # ACCEPT ASTRONAUT INPUTS OF TIG.DELV(LV)
7 # CALL IMU STATUS CHECK ROUTINE (R02)
8 # DISPLAY TIME TO GO, APOGEE, PERIGEE, DELV(MAG), MGA AT IGN
9 # REQUEST BURN PROGRAM
10 #
11 # CALLING SEQUENCE VIA JOB FROM V37
12 #
13 # EXIT VIA V37 CALL OR TO GOTOPOOH (V34E)
14 #
15 # SUBROUTINE CALLS-FLAGUP, PHASCHNG, BANKCALL, ENDOFJOB, GOFLASH, GOFLASHR
16 # GOPERF3R, INTPRET, BLANKET, GOTOPOOH, R02BOTH, S30.1,
17 # TIG/N35, MIDGIM, DISPMGA
18 #
19 # ERASABLE INITIALIZATION- STATE VECTOR
20 #
21 # OUTPUT-RINIT, VINIT, +MGA, VTIG, RTIG, DELVSIN, DELVSAB, DELVSLV, HAPO,
22 # HPER, TTOGO
23 #
24 # DEBRIS- A,L, MPAC, PUSHLIST
25
26 BANK 32
27 SETLOC P30S
28
29 BANK
30 EBANK= +MGA
31 COUNT* $$/P30
32
33 P30 TC UPFLAG # SET UPDATE FLAG
34 ADRES UPDATFLG
35 TC UPFLAG # SET TRACK FLAG
36 ADRES TRACKFLG
37
38 P30N33 CAF V06N33 # T OF IGN
39 TC VNPOOH # RETURNS ON PROCEED, POOH ON TERMINATE
40
41 CAF V06N81 # DISPLAY DELTA V (LV)
42 TC VNPOOH # REDISPLAY ON RECYCLE
43
44 TC DOWNFLAG # RESET UPDATE FLAG
45 ADRES UPDATFLG
46 TC INTPRET
47 CALL
48
49 SET S30.1
50 SET EXIT
51 SET UPDATFLG
52
53 PARAM30 CAF V06N42 # DISPLAY APOGEE,PERIGEE ,DELTA V
54 TC VNPOOH
```



|    |  |  |  |  |    |
|----|--|--|--|--|----|
| 1  |  |  |  |  | 1  |
| 2  |  |  |  |  | 2  |
| 3  |  |  |  |  | 3  |
| 4  |  |  |  |  | 4  |
| 5  |  |  |  |  | 5  |
| 6  |  |  |  |  | 6  |
| 7  |  |  |  |  | 7  |
| 8  |  |  |  |  | 8  |
| 9  |  |  |  |  | 9  |
| 10 |  |  |  |  | 10 |
| 11 |  |  |  |  | 11 |
| 12 |  |  |  |  | 12 |
| 13 |  |  |  |  | 13 |
| 14 |  |  |  |  | 14 |
| 15 |  |  |  |  | 15 |
| 16 |  |  |  |  | 16 |
| 17 |  |  |  |  | 17 |
| 18 |  |  |  |  | 18 |
| 19 |  |  |  |  | 19 |
| 20 |  |  |  |  | 20 |
| 21 |  |  |  |  | 21 |
| 22 |  |  |  |  | 22 |
| 23 |  |  |  |  | 23 |
| 24 |  |  |  |  | 24 |
| 25 |  |  |  |  | 25 |
| 26 |  |  |  |  | 26 |
| 27 |  |  |  |  | 27 |
| 28 |  |  |  |  | 28 |
| 29 |  |  |  |  | 29 |
| 30 |  |  |  |  | 30 |
| 31 |  |  |  |  | 31 |
| 32 |  |  |  |  | 32 |
| 33 |  |  |  |  | 33 |
| 34 |  |  |  |  | 34 |
| 35 |  |  |  |  | 35 |
| 36 |  |  |  |  | 36 |
| 37 |  |  |  |  | 37 |
| 38 |  |  |  |  | 38 |
| 39 |  |  |  |  | 39 |
| 40 |  |  |  |  | 40 |
| 41 |  |  |  |  | 41 |
| 42 |  |  |  |  | 42 |
| 43 |  |  |  |  | 43 |
| 44 |  |  |  |  | 44 |
| 45 |  |  |  |  | 45 |
| 46 |  |  |  |  | 46 |
| 47 |  |  |  |  | 47 |
| 48 |  |  |  |  | 48 |
| 49 |  |  |  |  | 49 |
| 50 |  |  |  |  | 50 |
| 51 |  |  |  |  | 51 |
| 52 |  |  |  |  | 52 |
| 53 |  |  |  |  | 53 |
| 54 |  |  |  |  | 54 |
| 55 |  |  |  |  | 55 |
| 56 |  |  |  |  | 56 |
| 57 |  |  |  |  | 57 |
| 58 |  |  |  |  | 58 |
| 59 |  |  |  |  | 59 |
| 60 |  |  |  |  | 60 |



```
1 # PROGRAM DESCRIPTION S30.1 DATE 9NOV66
2 # MOD NO 1 LOG SECTION P30,P37
3
4 # MOD BY RAMA AIYAWAR **
5 # FUNCTIONAL DESCRIPTION
6 # BASED ON STORED TARGET PARAMETERS(R OF IGNITION(RTIG),V OF
7 # IGNITION(VTIG),TIME OF IGNITION (TIG)),COMPUTE PERIGEE ALTITUDE
8 # APOGEE ALTITUDE AND DELTAV REQUIRED(DELV SIN).
9 # CALLING SEQUENCE
10 # L CALL
11 # L+1 S30.1
12 # NORMAL EXIT MODE
13 # AT L+2 OR CALLING SEQUENCE (GOTO L+2)
14 # SUBROUTINES CALLED
15 # LEMPREC
16 # PERIAPO
17 # ALARM OR ABORT EXIT MODES
18 # NONE
19 # ERASABLE INITIALIZATION REQUIRED
20 # TIG TIME OF IGNITION DP B28CS
21 # DELVSLV SPECIFIED DELTA-V IN LOCAL VERT.
22 # COORDS. OF ACTIVE VEHICLE AT
23 # TIME OF IGNITION VECTOR B+7 METERS/CS
24 #
25 # OUTPUT
26 # RTIG POSITION AT TIG VECTOR B+29 METERS
27 # VTIG VELOCITY AT TIG VECTOR B+29 METERS/CS
28 # PDL 4D APOGEE ALTITUDE DP B+29 M , B+27 METERS.
29 # HAPO APOGEE ALTITUDE DP B+29 METERS
30 # PDL 8D PERIGEE ALTITUDE DP B+29 M , B+27 METERS.
31 # HPER PERIGEE ALTITUDE DP B+29 METERS
32 # DELVSIN SPECIFIED DELTA-V IN INTERTIAL
33 # COORD. OF ACTIVE VEHICLE AT
34 # TIME OF IGNITION VECTOR B+7 METERS/CS
35 # DELVSAB MAG. OF DELVSIN VECTOR B+7 METERS/CS
36 #
37 # DEBRIS QTEMP TEMP. ERASABLE
38 # QPRET,MPAC
39 # PUSHLIST
40
41 # SETLOC P30S1
42 # BANK
43
44 # COUNT* $$/S30S
45
46 S30.1 STQ DLOAD
47 # QTEMP
48 # TIG # TIME IGNITION SCALED AT 2(+28)CS
49 # STCALL TDEC1
50 # LEMPREC # ENCKE ROUTINE FOR LEM
51
52 # VLOAD SXA,2
```

|    |           |        |          |                                            |    |
|----|-----------|--------|----------|--------------------------------------------|----|
| 1  | # 150-151 |        |          | PAGE 017                                   | 1  |
| 2  |           | RATT   |          |                                            | 2  |
| 3  |           | RTX2   |          |                                            | 3  |
| 4  |           | STORE  | RTIG     | # RADIUS VECTOR AT IGNITION TIME           | 5  |
| 5  |           | UNIT   | VCOMP    |                                            | 6  |
| 6  |           | STOVL  | DELVSIN  | # ZRF/LV IN DELVSIN SCALED AT 2            | 7  |
| 7  |           |        | VATT     | # VELOCITY VECTOR AT TIG, SCALED 2(7) M/CS | 9  |
| 8  |           | STORE  | VTIG     |                                            | 10 |
| 9  |           | VXV    | UNIT     |                                            | 11 |
| 10 |           |        | RTIG     |                                            | 13 |
| 11 |           | SETPD  | SXA,1    |                                            | 14 |
| 12 |           |        | 0        |                                            | 15 |
| 13 |           |        | RTX1     |                                            | 17 |
| 14 |           | PUSH   | VXV      | # YRF/LV PDL 0 SCALED AT 2                 | 18 |
| 15 |           |        | DELVSIN  |                                            | 19 |
| 16 |           | VSL1   | PDVL     |                                            | 21 |
| 17 |           | PDVL   | PDVL     | # YRF/LV PDL 6 SCALED AT 2                 | 22 |
| 18 |           |        | DELVSIN  | # ZRF/LV PDL 12D SCALED AT 2               | 23 |
| 19 |           |        | DELVSLV  |                                            | 24 |
| 20 |           | VXM    | VSL1     |                                            | 25 |
| 21 |           |        | 0        |                                            | 26 |
| 22 |           | STORE  | DELVSIN  | # DELTAV IN INERT. COOR. SCALED TO B+7M/CS | 27 |
| 23 |           | ABVAL  |          |                                            | 28 |
| 24 |           | STOVL  | DELVSAB  | # DELTA V MAG.                             | 29 |
| 25 |           |        | RTIG     | # (FOR PERIAPO)                            | 30 |
| 26 |           | PDVL   | VAD      | # VREQUIRED = VTIG + DELVSIN (FOR PERIAPO) | 31 |
| 27 |           |        | VTIG     |                                            | 32 |
| 28 |           |        | DELVSIN  |                                            | 33 |
| 29 |           | CALL   |          |                                            | 34 |
| 30 |           |        | PERIAPO1 |                                            | 35 |
| 31 |           | CALL   |          |                                            | 36 |
| 32 |           |        | SHIFTR1  | # RESCALE IF NEEDED                        | 37 |
| 33 |           | CALL   |          | # LIMIT DISPLAY TO 9999.9 N. MI.           | 38 |
| 34 |           |        | MAXCHK   |                                            | 39 |
| 35 |           | STODL  | HPER     | # PERIGEE ALT 2(29) METERS, FOR DISPLAY    | 40 |
| 36 |           |        | 4D       |                                            | 41 |
| 37 |           | CALL   |          |                                            | 42 |
| 38 |           |        | SHIFTR1  | # RESCALE IF NEEDED                        | 43 |
| 39 |           | CALL   |          | # LIMIT DISPLAY TO 9999.9 N. MI.           | 44 |
| 40 |           |        | MAXCHK   |                                            | 45 |
| 41 |           | STCALL | HAPO     | # APOGEE ALT 2(29) METERS, FOR DISPLAY     | 46 |
| 42 |           |        | QTEMP    |                                            | 47 |
| 43 |           |        |          |                                            | 48 |
| 44 |           |        |          |                                            | 49 |
| 45 |           |        |          |                                            | 50 |
| 46 |           |        |          |                                            | 51 |
| 47 |           |        |          |                                            | 52 |
| 48 |           |        |          |                                            | 53 |
| 49 |           |        |          |                                            | 54 |
| 50 |           |        |          |                                            | 55 |
| 51 |           |        |          |                                            | 56 |
| 52 |           |        |          |                                            | 57 |
| 53 |           |        |          |                                            | 58 |
| 54 |           |        |          |                                            | 59 |
| 55 |           |        |          |                                            | 60 |
| 56 |           |        |          |                                            | 61 |
| 57 |           |        |          |                                            | 62 |
| 58 |           |        |          |                                            | 63 |
| 59 |           |        |          |                                            | 64 |
| 60 |           |        |          |                                            | 65 |

## # COELLIPTIC SEQUENCE INITIATION (CSI) PROGRAMS (P32 AND P72)

# MOD NO -1 LOG SECTION -- P32-P35, P72-P75  
# MOD BY WHITE.P DATE 1JUNE67

## # PURPOSE

# (1) TO CALCULATE PARAMETERS ASSOCIATED WITH THE TIME FOLLOWING  
# CONCENTRIC FLIGHT PLAN MANEUVERS -- THE CO-ELLIPTIC SEQUENCE  
# INITIATION (CSI) MANEUVER AND THE CONSTANT DELTA ALTITUDE  
# (CDH) MANEUVER.

# (2) TO CALCULATE THESE PARAMETERS BASED UPON MANEUVER DATA  
# APPROVED AND KEYED INTO THE DSKY BY THE ASTRONAUT.

# (3) TO DISPLAY TO THE ASTRONAUT AND THE GROUND DEPENDENT VARIABLES  
# ASSOCIATED WITH THE CONCENTRIC FLIGHT PLAN MANEUVERS FOR  
# APPROVAL BY THE ASTRONAUT/GROUND.

# (4) TO STORE THE CSI TARGET PARAMETERS FOR USE BY THE DESIRED  
# THRUSTING PROGRAM.

## # ASSUMPTIONS

# (1) AT A SELECTED TPI TIME THE LINE OF SIGHT BETWEEN THE ACTIVE  
# AND PASSIVE VEHICLES IS SELECTED TO BE A PRESCRIBED ANGLE (E)  
# FROM THE HORIZONTAL PLANE DEFINED BY THE ACTIVE VEHICLE  
# POSITION.

# (2) THE TIME BETWEEN CSI IGNITION AND CDH IGNITION MUST BE  
# COMPUTED TO BE GREATER THAN 10 MINUTES FOR SUCCESSFUL  
# COMPLETION OF THE PROGRAM.

# (3) THE TIME BETWEEN CDH IGNITION AND TPI IGNITION MUST BE  
# COMPUTED TO BE GREATER THAN 10 MINUTES FOR SUCCESSFUL  
# COMPLETION OF THE PROGRAM.

# (4) CDH DELTA V IS SELECTED TO MINIMIZE THE VARIATION OF THE  
# ALTITUDE DIFFERENCE BETWEEN THE ORBITS.

# (5) CSI BURN IS DEFINED SUCH THAT THE IMPULSIVE DELTA V IS IN THE  
# HORIZONTAL PLANE DEFINED BY THE ACTIVE VEHICLE POSITION AT CSI  
# IGNITION.

# (6) THE PERICENTER ALTITUDE OF THE ORBIT FOLLOWING CSI AND CDH  
# MUST BE GREATER THAN 35,000 FT (LUNAR ORBIT) OR 85 NM (EARTH  
# ORBIT) FOR SUCCESSFUL COMPLETION OF THIS PROGRAM.

# (7) THE CSI AND CDH MANEUVERS ARE ORIGINALLY ASSUMED TO BE  
# PARALLEL TO THE PLANE OF THE CSM ORBIT. HOWEVER, CREW

|    |   |       |                                                                |    |
|----|---|-------|----------------------------------------------------------------|----|
| 1  | # |       | MODIFICATION OF DELTA V (LV) COMPONENTS MAY RESULT IN AN       | 1  |
| 2  | # |       | OUT-OF-PLANE CSI MANEUVER                                      | 2  |
| 3  |   |       |                                                                | 3  |
| 4  |   |       |                                                                | 4  |
| 5  | # | (8)   | STATE VECTOR UPDATES BY P27 ARE DISALLOWED DURING AUTOMATIC    | 5  |
| 6  | # |       | STATE VECTOR UPDATING INITIATED BY P20 (SEE ASSUMPTION 10).    | 6  |
| 7  |   |       |                                                                | 7  |
| 8  | # | (9)   | COMPUTED VARIABLES MAY BE STORED FOR LATER VERIFICATION BY     | 8  |
| 9  | # |       | THE GROUND. THESE STORAGE CAPABILITIES ARE NORMALLY LIMITED    | 9  |
| 10 | # |       | ONLY TO THE PARAMETERS FOR ONE THRUSTING MANEUVER AT A TIME    | 10 |
| 11 | # |       | EXCEPT FOR CONCENTRIC FLIGHT PLAN MANEUVER SEQUENCES.          | 11 |
| 12 |   |       |                                                                | 12 |
| 13 | # | (10)  | THE RENDEZVOUS RADAR MAY OR MAY NOT BE USED TO UPDATE THE LM   | 13 |
| 14 | # |       | OR CSM STATE VECTORS FOR THIS PROGRAM. IF RADAR USE IS         | 14 |
| 15 | # |       | DESIRED THE RADAR WAS TURNED ON AND LOCKED BY THE CSM BY       | 15 |
| 16 | # |       | PREVIOUS SELECTION OF P20. RADAR SIGHTING MARKS WILL BE MADE   | 16 |
| 17 | # |       | AUTOMATICALLY APPROXIMATELY ONCE A MINUTE WHEN ENABLED BY THE  | 17 |
| 18 | # |       | TRACK AND UPDATE FLAGS (SEE P20). THE RENDEZVOUS TRACKING      | 18 |
| 19 | # |       | MARK COUNTER IS ZEROED BY THE SELECTION OF P20 AND AFTER EACH  | 19 |
| 20 | # |       | THRUSTING MANEUVER.                                            | 20 |
| 21 |   |       |                                                                | 21 |
| 22 | # | (11)  | THE ISS NEED NOT BE ON TO COMPLETE THIS PROGRAM.               | 22 |
| 23 |   |       |                                                                | 23 |
| 24 | # | (12)  | THE OPERATION OF THE PROGRAM UTILIZES THE FOLLOWING FLAGS --   | 24 |
| 25 | # |       |                                                                | 25 |
| 26 | # |       | ACTIVE VEHICLE FLAG -- DESIGNATES THE VEHICLE WHICH IS         | 26 |
| 27 | # |       | DOING RENDEZVOUS THRUSTING MANEUVERS TO THE PROGRAM WHICH      | 27 |
| 28 | # |       | CALCULATES THE MANEUVER PARAMETERS. SET AT THE START OF        | 28 |
| 29 | # |       | EACH RENDEZVOUS PRE-THRUSTING PROGRAM.                         | 29 |
| 30 | # |       |                                                                | 30 |
| 31 | # |       | FINAL FLAG -- SELECTS FINAL PROGRAM DISPLAYS AFTER CREW HAS    | 31 |
| 32 | # |       | COMPLETED THE FINAL MANEUVER COMPUTATION AND DISPLAY           | 32 |
| 33 | # |       | CYCLE.                                                         | 33 |
| 34 | # |       |                                                                | 34 |
| 35 | # |       | EXTERNAL DELTA V STEERING FLAG -- DESIGNATES THE TYPE OF       | 35 |
| 36 | # |       | STEERING REQUIRED FOR EXECUTION OF THIS MANEUVER BY THE        | 36 |
| 37 | # |       | THRUSTING PROGRAM SELECTED AFTER COMPLETION OF THIS            | 37 |
| 38 | # |       | PROGRAM.                                                       | 38 |
| 39 | # |       |                                                                | 39 |
| 40 | # | (13)  | IT IS NORMALLY REQUIRED THAT THE ISS BE ON FOR 1 HOUR PRIOR TO | 40 |
| 41 | # |       | A THRUSTING MANEUVER.                                          | 41 |
| 42 | # |       |                                                                | 42 |
| 43 | # | (14)  | THIS PROGRAM IS SELECTED BY THE ASTRONAUT BY DSKY ENTRY        | 43 |
| 44 | # |       |                                                                | 44 |
| 45 | # |       | P32 IF THIS VEHICLE IS ACTIVE VEHICLE.                         | 45 |
| 46 | # |       |                                                                | 46 |
| 47 | # |       | P72 IF THIS VEHICLE IS THE PASSIVE VEHICLE.                    | 47 |
| 48 | # |       |                                                                | 48 |
| 49 | # | INPUT |                                                                | 49 |
| 50 |   |       |                                                                | 50 |
| 51 | # | (1)   | TCSI TIME OF THE CSI MANEUVER                                  | 51 |
| 52 |   |       |                                                                | 52 |
| 53 |   |       |                                                                | 53 |
| 54 |   |       |                                                                | 54 |
| 55 |   |       |                                                                | 55 |
| 56 |   |       |                                                                | 56 |
| 57 |   |       |                                                                | 57 |
| 58 |   |       |                                                                | 58 |
| 59 |   |       |                                                                | 59 |
| 60 |   |       |                                                                | 60 |

1412THE

```
1 # (2) NN NUMBER OF APSIDAL CROSSINGS THRU WHICH THE ACTIVE
2 # (2) NN VEHICLE ORBIT CAN BE ADVANCED TO OBTAIN THE CDH
3 # (2) NN MANEUVER POINT.
4 # (3) ELEV DESIRED LOS ANGLE AT TPI
5 # (4) TTPI TIME OF THE TPI MANEUVER
6 #
7 # OUTPUT
8
9
10 # (1) TRMKMCNT NUMBER OF MARKS
11 # (2) TTOGO TIME TO GO
12 # (3) +MGA MIDDLE GIMBAL ANGLE
13 # (4) DIFFALT DELTA ALTITUDE AT CDH
14 # (5) T1TOT2 DELTA TIME FROM CSI TO CDH
15 # (6) T2TOT3 DELTA TIME FROM CDH TO TPI
16 # (7) DELVLVC DELTA VELOCITY AT CSI -- LOCAL VERTICAL COORDINATES
17 # (8) DELVLVC DELTA VELOCITY AT CDH -- LOCAL VERTICAL COORDINATES
18 #
19 # DOWNLINK
20
21 # (1) TCSI TIME OF THE CSI MANEUVER
22 # (2) TCDH TIME OF THE CDH MANEUVER
23 # (3) TTPI TIME OF THE TPI MANEUVER
24 # (4) TIG TIME OF THE CSI MANEUVER
25 # (5) DELVEET1 DELTA VELOCITY AT CSI -- REFERENCE COORDINATES
26 # (6) DELVEET2 DELTA VELOCITY AT CDH -- REFERENCE COORDINATES
27 # (7) DIFFALT DELTA ALTITUDE AT CDH
28 # (8) NN NUMBER OF APSIDAL CROSSINGS THRU WHICH THE ACTIVE
29 # (8) NN VEHICLE ORBIT CAN BE ADVANCED TO OBTAIN THE CDH
30 # (8) NN MANEUVER POINT
31 # (9) ELEV DESIRED LOS ANGLE AT TPI
32 #
33 # COMMUNICATION TO THRUSTING PROGRAM
34
35 # (1) TIG TIME OF THE CSI MANEUVER
36 # (2) RTIG POSITION OF ACTIVE VEHICLE AT CSI -- BEFORE ROTATION
37 # (2) RTIG INTO PLANE OF PASSIVE VEHICLE
38 # (3) VTIG VELOCITY OF ACTIVE VEHICLE AT CSE -- BEFORE ROTATION
39 # (3) VTIG INTO PLANE OF PASSIVE VEHICLE
40 # (4) DELVSIN DELTA VELOCITY AT CSI -- REFERENCE COORDINATES
41 # (5) DELVSAB MAGNITUDE OF DELTA VELOCITY AT CSI
42 # (6) XDELVFLG SET TO INDICATE EXTERNAL DELTA V VG COMPUTATION
43 #
44 # SUBROUTINES USED
45
46 # AVFLAGA
47 # AVFLAGP
48 # P20FLGON
49 # VARALARM
50 # BANKCALL
51 # GOFLASH
52 # GOTOPOOH
```

# P32-P35\_P72-P75

|    |          |          |               |  |  |
|----|----------|----------|---------------|--|--|
| 1  |          |          |               |  |  |
| 2  | #        | VNPOOH   |               |  |  |
| 3  | #        | GOFLASHR |               |  |  |
| 4  | #        | BLANKET  |               |  |  |
| 5  | #        | ENDOFJOB |               |  |  |
| 6  | #        | SELECTMU |               |  |  |
| 7  | #        | ADVANCE  |               |  |  |
| 8  | #        | INTINT   |               |  |  |
| 9  | #        | PASSIVE  |               |  |  |
| 10 | #        | CSI/A    |               |  |  |
| 11 | #        | S32/33.1 |               |  |  |
| 12 | #        | DISDVLVC |               |  |  |
| 13 | #        | VN1645   |               |  |  |
| 14 |          |          |               |  |  |
| 15 |          | BANK     | 35            |  |  |
| 16 |          | SETLOC   | CSI/CDH       |  |  |
| 17 |          | BANK     |               |  |  |
| 18 |          | EBANK=   | SUBEXIT       |  |  |
| 19 |          | COUNT*   | \$\$/P3272    |  |  |
| 20 | P32      | TC       | AVFLAGA       |  |  |
| 21 |          | TC       | P32STRT       |  |  |
| 22 | P72      | TC       | AVFLAGP       |  |  |
| 23 | P32STRT  | EXTEND   |               |  |  |
| 24 |          | DCA      | P30ZERO       |  |  |
| 25 |          | DXCH     | CENTANG       |  |  |
| 26 |          | TC       | P32/P72A      |  |  |
| 27 | ALMXITA  | SXA,2    |               |  |  |
| 28 |          |          | CSIALRM       |  |  |
| 29 | ALMXIT   | LXC,1    |               |  |  |
| 30 |          |          | CSIALRM       |  |  |
| 31 |          | SLOAD*   | EXIT          |  |  |
| 32 |          |          | ALARM/TB -1,1 |  |  |
| 33 |          | CA       | MPAC          |  |  |
| 34 |          | TC       | VARALARM      |  |  |
| 35 |          | CAF      | V05N09        |  |  |
| 36 |          | TC       | BANKCALL      |  |  |
| 37 |          | CADR     | GOFLASH       |  |  |
| 38 |          | TC       | GOTOPPOH      |  |  |
| 39 |          | TC       | -4            |  |  |
| 40 | P32/P72A | TC       | P20FLGON      |  |  |
| 41 |          | CAF      | P30ZERO       |  |  |
| 42 |          | TS       | NN +1         |  |  |
| 43 |          | TS       | TCSI          |  |  |
| 44 |          | TS       | TCSI +1       |  |  |
| 45 | VN0611   | CAF      | V06N11 # TCSI |  |  |
| 46 |          | TC       | VNPOOH        |  |  |
| 47 |          | TC       | INTPRET       |  |  |
| 48 |          | DLOAD    | DCOMP         |  |  |
| 49 |          |          | TCSI          |  |  |
| 50 |          | BMN      | DLOAD         |  |  |
| 51 |          |          | VN0655        |  |  |
| 52 |          |          |               |  |  |
| 53 |          |          |               |  |  |
| 54 |          |          |               |  |  |
| 55 |          |          |               |  |  |
| 56 |          |          |               |  |  |
| 57 |          |          |               |  |  |
| 58 |          |          |               |  |  |
| 59 |          |          |               |  |  |
| 60 |          |          |               |  |  |

|    |            |        |           |    |
|----|------------|--------|-----------|----|
| 1  | # P32/P72B |        |           | 1  |
| 2  |            |        | TETLEM    | 2  |
| 3  |            | STCALL | TDEC1     | 3  |
| 4  |            |        | PRECSET   | 4  |
| 5  |            | VLOAD  | VSR*      | 5  |
| 6  |            |        | RACT3     | 6  |
| 7  |            |        | 0,2       | 7  |
| 8  |            | STOVL  | RVEC      | 8  |
| 9  |            |        | VACT3     | 9  |
| 10 |            | VSR*   | SET       | 10 |
| 11 |            |        | 0,2       | 11 |
| 12 |            |        | RVSW      | 12 |
| 13 |            | STODL  | VVEC      | 13 |
| 14 |            |        | DPPOS MAX | 14 |
| 15 |            | STCALL | RDESIRED  | 15 |
| 16 |            |        | TIMERAD   | 16 |
| 17 |            | DAD    |           | 17 |
| 18 |            |        | TDEC2     | 18 |
| 19 |            | STORE  | TCSI      | 19 |
| 20 |            | EXIT   |           | 20 |
| 21 |            | TC     | VN0611    | 21 |
| 22 | VN0655     | EXIT   |           | 22 |
| 23 |            | CAF    | V06N55    | 23 |
| 24 |            | TC     | BANKCALL  | 24 |
| 25 |            | CADR   | GOFLASH   | 25 |
| 26 |            | TC     | GOTOP00H  | 26 |
| 27 |            | TC     | +2        | 27 |
| 28 |            | TC     | -5        | 28 |
| 29 |            | CAF    | V06N37    | 29 |
| 30 |            | TC     | VNPOOH    | 30 |
| 31 |            | TC     | INTPRET   | 31 |
| 32 |            | DLOAD  |           | 32 |
| 33 |            |        | TCSI      | 33 |
| 34 |            | STCALL | TIG       | 34 |
| 35 |            |        | SELECTMU  | 35 |
| 36 | P32/P72B   | CALL   |           | 36 |
| 37 |            |        | ADVANCE   | 37 |
| 38 |            | SETPD  | VLOAD     | 38 |
| 39 |            |        | OD        | 39 |
| 40 |            |        | VPASS1    | 40 |
| 41 |            | PDVL   | PDDL      | 41 |
| 42 |            |        | RPASS1    | 42 |
| 43 |            |        | TCSI      | 43 |
| 44 |            | PDDL   | PDDL      | 44 |
| 45 |            |        | TTPI      | 45 |
| 46 |            |        | TWOPI     | 46 |
| 47 |            | PUSH   | CALL      | 47 |
| 48 |            |        | INTINT    | 48 |
| 49 |            | CALL   |           | 49 |
| 50 |            |        | PASSIVE   | 50 |
| 51 |            | CALL   |           | 51 |
| 52 |            |        |           | 52 |
| 53 |            |        |           | 53 |
| 54 |            |        |           | 54 |
| 55 |            |        |           | 55 |
| 56 |            |        |           | 56 |
| 57 |            |        |           | 57 |
| 58 |            |        |           | 58 |
| 59 |            |        |           | 59 |
| 60 |            |        |           | 60 |

|    |            |       |          |    |
|----|------------|-------|----------|----|
| 1  | # P32/P72C |       | CSI/A    | 1  |
| 2  |            |       | SET      | 2  |
| 3  | P32/P72C   | BON   | FINALFLG | 3  |
| 4  |            |       | P32/P72D | 4  |
| 5  |            |       | UPDATFLG | 5  |
| 6  |            |       |          | 6  |
| 7  | P32/P72D   | DLOAD |          | 7  |
| 8  |            |       | T1TOT2   | 8  |
| 9  | P32/P72E   | STORE | T1TOT2   | 9  |
| 10 |            |       | BPL      | 10 |
| 11 |            |       | 60MIN    | 11 |
| 12 |            |       | P32/P72E | 12 |
| 13 |            |       |          | 13 |
| 14 |            |       | T2TOT3   | 14 |
| 15 | P32/P72F   | STORE | T2TOT3   | 15 |
| 16 |            |       | BPL      | 16 |
| 17 |            |       | 60MIN    | 17 |
| 18 |            |       | P32/P72F | 18 |
| 19 |            |       |          | 19 |
| 20 |            |       | V06N75   | 20 |
| 21 |            |       | VNP00H   | 21 |
| 22 |            |       | INTPRET  | 22 |
| 23 |            |       | CALL     | 23 |
| 24 |            |       | DELVEET1 | 24 |
| 25 |            |       | S32/33.1 | 25 |
| 26 |            |       | DELVEET1 | 26 |
| 27 |            |       | RACT2    | 27 |
| 28 |            |       | RACT1    | 28 |
| 29 |            |       | DELVEET2 | 29 |
| 30 |            |       | CALL     | 30 |
| 31 |            |       | 0682     | 31 |
| 32 |            |       | DISDVLVC | 32 |
| 33 |            |       |          | 33 |
| 34 |            |       | TTPI     | 34 |
| 35 |            |       | TTPIO    | 35 |
| 36 |            |       | VN1645   | 36 |
| 37 |            |       |          | 37 |
| 38 |            |       | P32/P72B | 38 |
| 39 |            |       |          | 39 |
| 40 |            |       |          | 40 |
| 41 |            |       |          | 41 |
| 42 |            |       |          | 42 |
| 43 |            |       |          | 43 |
| 44 |            |       |          | 44 |
| 45 |            |       |          | 45 |
| 46 |            |       |          | 46 |
| 47 |            |       |          | 47 |
| 48 |            |       |          | 48 |
| 49 |            |       |          | 49 |
| 50 |            |       |          | 50 |
| 51 |            |       |          | 51 |
| 52 |            |       |          | 52 |
| 53 |            |       |          | 53 |
| 54 |            |       |          | 54 |
| 55 |            |       |          | 55 |
| 56 |            |       |          | 56 |
| 57 |            |       |          | 57 |
| 58 |            |       |          | 58 |
| 59 |            |       |          | 59 |
| 60 |            |       |          | 60 |



# CONSTANT DELTA HEIGHT (CDH) PROGRAMS (P33 AND P73)  
# MOD NO -1 LOC SECTION -- P32-P35, P72-P75  
# MOD BY WHITE, P. DATE: 1 JUNE 67

## # PURPOSE

- # (1) TO CALCULATE PARAMETERS ASSOCIATED WITH THE CONSTANT DELTA ALTITUDE MANEUVER (CDH).
- # (2) TO CALCULATE THESE PARAMETERS BASED UPON MANEUVER DATA APPROVED AND KEYED INTO THE DSKY BY THE ASTRONAUT.
- # (3) TO DISPLAY TO THE ASTRONAUT AND THE GROUND DEPENDENT VARIABLES ASSOCIATED WITH THE CDH MANEUVER FOR APPROVAL BY THE ASTRONAUT/GROUND.
- # (4) TO STORE THE CDH TARGET PARAMETERS FOR USE BY THE DESIRED THRUSTING PROGRAM.

## # ASSUMPTIONS

- # (1) THIS PROGRAM IS BASED UPON PREVIOUS COMPLETION OF THE CO-ELLIPTIC SEQUENCE INITIATION (CSI) PROGRAM (P32/P72).  
# THEREFORE --
  - # (A) AT A SELECTED TPI TIME (NOW IN STORAGE) THE LINE OF SIGHT BETWEEN THE ACTIVE AND PASSIVE VEHICLES WAS SELECTED TO BE A PRESCRIBED ANGLE (E) (NOW IN STORAGE) FROM THE HORIZONTAL PLANE DEFINED BY THE ACTIVE VEHICLE POSITION.
  - # (B) THE TIME BETWEEN CSI IGNITION AND CDH IGNITION WAS COMPUTED TO BE GREATER THAN 10 MINUTES.
  - # (C) THE TIME BETWEEN CDH IGNITION AND TPI IGNITION WAS COMPUTED TO BE GREATER THAN 10 MINUTES.
  - # (D) THE VARIATION OF THE ALTITUDE DIFFERENCE BETWEEN THE ORBITS WAS MINIMIZED.
  - # (E) CSI BURN WAS DEFINED SUCH THAT THE IMPULSIVE DELTA V WAS IN THE HORIZONTAL PLANE DEFINED BY ACTIVE VEHICLE POSITION AT CSI IGNITION.
  - # (F) THE PERICENTER ALTITUDES OF THE ORBITS FOLLOWING CSI AND CDH WERE COMPUTED TO BE GREATER THAN 35,000 FT FOR LUNAR ORBIT OR 85 NM FOR EARTH ORBIT.
  - # (G) THE CSI AND CDH MANEUVERS WERE ASSUMED TO BE PARALLEL TO THE PLANE OF THE PASSIVE VEHICLE ORBIT. HOWEVER, CREW

MODIFICATION OF DELTA V (LV) COMPONENTS MAY HAVE RESULTED  
IN AN OUT-OF-PLANE MANEUVER.

(2) STATE VECTOR UPDATES BY P27 ARE DISALLOWED DURING AUTOMATIC  
STATE VECTOR UPDATING INITIATED BY P20 (SEE ASSUMPTION 4).

(3) COMPUTED VARIABLES MAY BE STORED FOR LATER VERIFICATION BY  
THE GROUND. THESE STORAGE CAPABILITIES ARE NORMALLY LIMITED  
ONLY TO THE PARAMETERS FOR ONE THRUSTING MANEUVER AT A TIME  
EXCEPT FOR CONCENTRIC FLIGHT PLAN MANEUVER SEQUENCES.

(4) THE RENDEZVOUS RADAR MAY OR MAY NOT BE USED TO UPDATE THE LM.  
OR CSM STATE VECTORS FOR THIS PROGRAM. IF RADAR USE IS  
DESIRED THE RADAR WAS TURNED ON AND LOCKED ON THE CSM BY  
PREVIOUS SELECTION OF P20. RADAR SIGHTING MARKS WILL BE MADE  
AUTOMATICALLY APPROXIMATELY ONCE A MINUTE WHEN ENABLED BY THE  
TRACK AND UPDATE FLAGS (SEE P20). THE RENDEZVOUS TRACKING  
MARK COUNTER IS ZEROED BY THE SELECTION OF P20 AND AFTER EACH  
THRUSTING MANEUVER.

(5) THE ISS NEED NOT BE ON TO COMPLETE THIS PROGRAM.

(6) THE OPERATION OF THE PROGRAM UTILIZES THE FOLLOWING FLAGS --

ACTIVE VEHICLE FLAG -- DESIGNATES THE VEHICLE WHICH IS  
DOING RENDEZVOUS THRUSTING MANEUVERS TO THE PROGRAM WHICH  
CALCULATES THE MANEUVER PARAMETERS. SET AT THE START OF  
EACH RENDEZVOUS PRE-THRUSTING PROGRAM.

FINAL FLAG -- SELECTS FINAL PROGRAM DISPLAYS AFTER CREW HAS  
COMPLETED THE FINAL MANEUVER COMPUTATION AND DISPLAY  
CYCLE.

EXTERNAL DELTA V STEERING FLAG -- DESIGNATES THE TYPE OF  
STEERING REQUIRED FOR EXECUTION OF THIS MANEUVER BY THE  
THRUSTING PROGRAM SELECTED AFTER COMPLETION OF THIS  
PROGRAM.

(7) IT IS NORMALLY REQUIRED THAT THE ISS BE ON FOR 1 HOUR PRIOR TO  
A THRUSTING MANEUVER.

(8) THIS PROGRAM IS SELECTED BY THE ASTRONAUT BY DSKY ENTRY.

P33 IF THIS VEHICLE IS ACTIVE VEHICLE.

P73 IF THIS VEHICLE IS PASSIVE VEHICLE.

# INPUT

(1) TTPIO TIME OF THE TPI MANEUVER -- SAVED FROM P32/P72

```
1 # P32-P35_P72-P75
2 # (2) ELEV DESIRED LOS ANGLE AT TPI -- SAVED FROM P32/P72
3 # (3) TCDH TIME OF THE CDH MANEUVER
4
5 # OUTPUT
6 #
7 # (1) TRMKCNT NUMBER OF MARKS
8 # (2) TTOGO TIME TO GO
9 # (3) +MGA MIDDLE GIMBAL ANGLE
10 # (4) DIFFALT DELTA ALTITUDE AT CDH
11 # (5) T2TOT3 DELTA TIME FROM CDH TO COMPUTED TPI
12 # (6) NOMTPI DELTA TIME FROM NOMINAL TPI TO COMPUTED TPI
13 # (7) DELVLVC DELTA VELOCITY AT CDH -- LOCAL VERTICAL COORDINATES
14 #
15 # DOWNLINK
16 #
17 # (1) TCDH TIME OF THE CDH MANEUVER
18 # (2) TTPI TIME OF THE TPI MANEUVER
19 # (3) TIG TIME OF THE CDH MANEUVER
20 # (4) DELLVEET2 DELTA VELOCITY AT CDH -- REFERENCE COORDINATES
21 # (5) DIFFALT DELTA ALTITUDE AT CDH
22 # (6) ELEV DESIRED LOS ANGLE AT TPI
23 #
24 # COMMUNICATION TO THRUSTING PROGRAMS
25 #
26 # (1) TIG TIME OF THE CDH MANEUVER
27 # (2) RTIG POSITION OF ACTIVE VEHICLE AT CDH -- BEFORE ROTATION
28 # INTO PLANE OF PASSIVE VEHICLE.
29 # (3) VTIG VELOCITY OF ACTIVE VEHICLE AT CDH -- BEFORE ROTATION
30 # INTO PLANE OF PASSIVE VEHICLE.
31 # (4) DELVSIN DELTA VELOCITY AT CDH -- REFERENCE COORDINATES.
32 # (5) DELVSAB MAGNITUDE OF DELTA VELOCITY AT CDH.
33 # (6) XDELVFLG SET TO INDICATE EXTERNAL DELTA V VG COMPUTATION.
34 #
35 # SUBROUTINES USED
36 #
37 # AVFLAGA
38 # AVFLAGP
39 # P20FLGON
40 # VNPOOH
41 # SELECTMU
42 # ADVANCE
43 # CDHMVR
44 # INTINT3P
45 # ACTIVE
46 # PASSIVE
47 # S33/S34.1
48 # ALARM
49 # BANKCALL
50 # GOFLASH
51 # GOTOPPOH
52 # S32/33.1
```

|    |                 |        |            |    |
|----|-----------------|--------|------------|----|
| 1  |                 |        |            | 1  |
| 2  | #               | VN1645 |            | 2  |
| 3  |                 |        |            | 3  |
| 4  | P33             | COUNT* | \$\$/P3373 | 4  |
| 5  |                 | TC     | AVFLAGA    | 5  |
| 6  |                 | TC     | P33/P73A   | 6  |
| 7  | P73<br>P33/P73A | TC     | AVFLAGP    | 7  |
| 8  |                 | TC     | P20FLGON   | 8  |
| 9  |                 | CAF    | V06N13     | 9  |
| 10 |                 |        |            | 10 |
| 11 |                 |        |            | 11 |
| 12 |                 |        |            | 12 |
| 13 |                 |        |            | 13 |
| 14 |                 |        |            | 14 |
| 15 |                 |        |            | 15 |
| 16 |                 |        |            | 16 |
| 17 | P33/P73B        | STCALL | TIG        | 17 |
| 18 |                 | CALL   | SELECTMU   | 18 |
| 19 |                 | CALL   | ADVANCE    | 19 |
| 20 |                 |        |            | 20 |
| 21 |                 |        |            | 21 |
| 22 |                 |        |            | 22 |
| 23 |                 |        |            | 23 |
| 24 |                 |        |            | 24 |
| 25 |                 |        |            | 25 |
| 26 |                 |        |            | 26 |
| 27 |                 |        |            | 27 |
| 28 |                 |        |            | 28 |
| 29 |                 |        |            | 29 |
| 30 |                 |        |            | 30 |
| 31 |                 |        |            | 31 |
| 32 |                 |        |            | 32 |
| 33 |                 |        |            | 33 |
| 34 |                 |        |            | 34 |
| 35 |                 |        |            | 35 |
| 36 |                 |        |            | 36 |
| 37 |                 |        |            | 37 |
| 38 |                 |        |            | 38 |
| 39 |                 |        |            | 39 |
| 40 |                 |        |            | 40 |
| 41 |                 |        |            | 41 |
| 42 |                 |        |            | 42 |
| 43 |                 |        |            | 43 |
| 44 |                 |        |            | 44 |
| 45 |                 |        |            | 45 |
| 46 |                 |        |            | 46 |
| 47 |                 |        |            | 47 |
| 48 |                 |        |            | 48 |
| 49 |                 |        |            | 49 |
| 50 |                 |        |            | 50 |
| 51 |                 |        |            | 51 |
| 52 |                 |        |            | 52 |
| 53 |                 |        |            | 53 |
| 54 |                 |        |            | 54 |
| 55 |                 |        |            | 55 |
| 56 |                 |        |            | 56 |
| 57 |                 |        |            | 57 |
| 58 |                 |        |            | 58 |
| 59 |                 |        |            | 59 |
| 60 |                 |        |            | 60 |

|    |          |        |          |    |
|----|----------|--------|----------|----|
| 1  |          |        |          | 1  |
| 2  |          | TC     | P33/P73A | 2  |
| 3  |          | TC     | INTPRET  | 3  |
| 4  |          | DLOAD  |          | 4  |
| 5  |          |        | P30ZERO  | 5  |
| 6  |          | STORE  | NOMTPI   | 6  |
| 7  | P33/P73C | BON    | SET      | 7  |
| 8  |          |        | FINALFLG | 8  |
| 9  |          |        | P33/P73D | 9  |
| 10 |          |        | UPDATFLG | 10 |
| 11 | P33/P73D | DLOAD  | DAD      | 11 |
| 12 |          |        | NOMTPI   | 12 |
| 13 |          |        | TTPI     | 13 |
| 14 |          | STORE  | TTPI     | 14 |
| 15 |          | DSU    |          | 15 |
| 16 |          |        | TCDH     | 16 |
| 17 | P33/P73E | DSU    | BPL      | 17 |
| 18 |          |        | 60MIN    | 18 |
| 19 |          |        | P33/P73E | 19 |
| 20 |          | DAD    |          | 20 |
| 21 |          |        | 60MIN    | 21 |
| 22 |          | STODL  | T1TOT2   | 22 |
| 23 |          |        | TTPI     | 23 |
| 24 |          | DSU    | PUSH     | 24 |
| 25 |          |        | TTPIO    | 25 |
| 26 | P33/P73F | ABS    | DSU      | 26 |
| 27 |          |        | 60MIN    | 27 |
| 28 |          | BPL    | DAD      | 28 |
| 29 |          |        | P33/P73F | 29 |
| 30 |          |        | 60MIN    | 30 |
| 31 |          | SIGN   | STADR    | 31 |
| 32 |          | STORE  | T2TOT3   | 32 |
| 33 |          | EXIT   |          | 33 |
| 34 |          | CAF    | V06N75   | 34 |
| 35 |          | TC     | VNPOOH   | 35 |
| 36 |          | TC     | INTPRET  | 36 |
| 37 |          | VLOAD  | CALL     | 37 |
| 38 |          |        | DELVEET2 | 38 |
| 39 |          |        | S32/33.1 | 39 |
| 40 |          | STCALL | DELVEET2 | 40 |
| 41 |          |        | VN1645   | 41 |
| 42 |          | GOTO   |          | 42 |
| 43 |          |        | P33/P73B | 43 |
| 44 |          |        |          | 44 |
| 45 |          |        |          | 45 |
| 46 |          |        |          | 46 |
| 47 |          |        |          | 47 |
| 48 |          |        |          | 48 |
| 49 |          |        |          | 49 |
| 50 |          |        |          | 50 |
| 51 |          |        |          | 51 |
| 52 |          |        |          | 52 |
| 53 |          |        |          | 53 |
| 54 |          |        |          | 54 |
| 55 |          |        |          | 55 |
| 56 |          |        |          | 56 |
| 57 |          |        |          | 57 |
| 58 |          |        |          | 58 |
| 59 |          |        |          | 59 |
| 60 |          |        |          | 60 |

```
SUBROUTINES USED
```

```
UPFLAG
```

| # | DOWNFLAG |
|---|----------|
|---|----------|

```
AVFLAGA EXTEND # AVFLAG = LEM
```

|      |         |
|------|---------|
| QXCH | SUBEXIT |
|------|---------|

|    |        |
|----|--------|
| TC | UPFLAG |
|----|--------|

| ADRES | AVFLAG |
|-------|--------|
|-------|--------|

```
TC SUBEXIT
```

```
AVFLAGP EXTEND # AVFLAG = CSM
```

|      |         |
|------|---------|
| QXCH | SUBEXIT |
|------|---------|

|    |          |
|----|----------|
| TC | DOWNFLAG |
|----|----------|

ADRES      AVFLAG

|    |         |
|----|---------|
| TC | SUBEXIT |
|----|---------|

P20FLGON      EXTEND

QXCH      SUBEXIT

| TC | UPFLAG |
|----|--------|
|----|--------|

ADRES      UPDATFLG                      # SET UPDATFLG

TC                      UPFLAG

| ADRES | TRACKFLG | # | SET | TRACKFLG |
|-------|----------|---|-----|----------|
|-------|----------|---|-----|----------|

```
TC SUBEXIT
```

# \*\*\*\*\* DISDVLVC \*\*\*\*\*

#

# SUBROUTINES USED

#

# S32/33.X

# VNPOOH

DISDVLVC

STORE

DELVLVC

STQ

CALL

NORMEX

S32/33.X

VLOAD

MXV

DELVLVC

OD

VSL1

SXA,1

VERBNOUN

STORE

DELVLVC

EXIT

CA

VERBNOUN

TC

VNPOOH

TC

INTPRET

GOTO

NORMEX

# \*\*\*\*\* CONSTANTS \*\*\*\*\*

|        |      |             |
|--------|------|-------------|
| V06N11 | VN   | 0611        |
| V06N13 | VN   | 0613        |
| V06N75 | VN   | 0675        |
| SN359+ | 2DEC | -.000086601 |
| CS359+ | 2DEC | +.499999992 |

|         |      |   |
|---------|------|---|
| P30ZERO | 2DEC | 0 |
|---------|------|---|

|       |      |        |
|-------|------|--------|
| 60MIN | 2DEC | 360000 |
|-------|------|--------|

|          |     |       |        |
|----------|-----|-------|--------|
| ALARM/TB | OCT | 00600 | # NO 1 |
|          | OCT | 00601 | # 2    |
|          | OCT | 00602 | # 3    |
|          | OCT | 00603 | # 4    |
|          | OCT | 00604 | # 5    |
|          | OCT | 00605 | # 6    |
|          | OCT | 00606 | # 7    |



# \*\*\*\*\* CSI/A \*\*\*\*\*

#

# SUBROUTINES USED

#

# VECSHIFT

# TIMETHET

# PERIAPO

# SHIFTR1

# INTINT2C

# CDHMVR

# PERIAPO1

# INTINT

# ACTIVE

BANK 34  
SETLOC CSI/CDH1  
BANK

LOOPMX EBANK= SUBEXIT  
COUNT\* \$\$/CSI  
2DEC 16

INITST 2DEC .03048 B-7 # INITIAL DELDV = 10 FPS

DVMAX1 2DEC 3.0480 B-7 # MAXIMUM DV1 = 1000 FPS

DVMAX2 2DEC 3.014472 B-7 # 989 FPS

1DPB2 2DEC 1.0 B-2

1DPB28 2DEC 1

PMINE 2DEC 157420 B-29 # 85 NM -- MUST BE 8 WORDS BEFORE PMINM

EPSILN1 2DEC .0003048 B-7 # .1 FPS

NICKELDP 2DEC .021336 B-7 # 7 FPS (CHANGED FROM .05 FPS)

FIFPSDP 2DEC -.152400 B-7 # 50 FPS

PMINM 2DEC 10668 B-29 # 35000 FT -- MUST BE 8 WORDS AFTER PMINE

DELMAX1 2DEC .6096000 B-7 # 200 FPS

ONETHTH 2DEC .0001 B-3

TMIN 2DEC 60000 # 10 MIN

CSI/A CLEAR SET # INITIALIZE INDICATORS  
S32.1F1 # DVT1 HAS EXCEEDED MAX INDICATOR  
S32.1F2 # FIRST PASS FOR NEWTON ITERATION INDICATOR

|    |                   |       |          |                                             |                   |    |
|----|-------------------|-------|----------|---------------------------------------------|-------------------|----|
| 1  | # FSE F32 F7E F7D |       |          | PAGE 055                                    |                   | 1  |
| 2  |                   | CLEAR | SET      |                                             |                   | 2  |
| 3  |                   |       | S32.1F3A | # 00=1ST 2 PASSES 2ND CYCLE, 01=FIRST CYCLE |                   | 3  |
| 4  |                   |       | S32.1F3B | # 10=2ND CYCLE, 11=50 FPS STAGE 2ND CYCLE   |                   | 4  |
| 5  |                   | DLOAD | P30ZERO  |                                             |                   | 5  |
| 6  |                   |       | LOOPCT   |                                             |                   | 6  |
| 7  | CSI/B             | STORE | CSIALRM  |                                             |                   | 7  |
| 8  |                   | STORE | VLOAD    |                                             |                   | 8  |
| 9  |                   | SETPD |          |                                             |                   | 9  |
| 10 |                   |       | OD       |                                             |                   | 10 |
| 11 |                   |       | RACT1    |                                             |                   | 11 |
| 12 |                   | ABVAL | PUSH     | # RA1                                       | B29 PL02D         | 12 |
| 13 |                   | NORM  | SR1      |                                             |                   | 13 |
| 14 |                   |       | X2       | #                                           | B29-N2+ B1 PL04D  | 14 |
| 15 |                   | PDVL  | ABVAL    |                                             |                   | 15 |
| 16 |                   |       | RPASS3   |                                             |                   | 16 |
| 17 |                   | NORM  | BDDV     | # RA1/RP3                                   | B1 PL02D          | 17 |
| 18 |                   |       | X1       |                                             |                   | 18 |
| 19 |                   | XSU,2 | SR*      | #                                           | B2                | 19 |
| 20 |                   |       | X1       |                                             |                   | 20 |
| 21 |                   |       | 1,2      |                                             |                   | 21 |
| 22 |                   | DAD   | DMP      | # (1+(RA1/RP3))RA1                          | B29+B2=B31 PL00D  | 22 |
| 23 |                   |       | 1DPB2    |                                             |                   | 23 |
| 24 |                   | NORM  | PDDL     | #                                           | PL02D             | 24 |
| 25 |                   |       | X1       |                                             |                   | 25 |
| 26 |                   |       | RTMU     |                                             |                   | 26 |
| 27 |                   | SR1   | DDV      | #                                           | B38-B31= B7 PL00D | 27 |
| 28 |                   | SL*   | SQRT     | #                                           | B7                | 28 |
| 29 |                   |       | 0        | -7,1                                        |                   | 29 |
| 30 |                   | PDVL  | UNIT     | #                                           | PL02D             | 30 |
| 31 |                   |       | RACT1    |                                             |                   | 31 |
| 32 |                   | PDVL  | VXV      |                                             |                   | 32 |
| 33 |                   |       | UP1      |                                             |                   | 33 |
| 34 |                   | UNIT  |          | # UNIT(URP1 X UVP1 X URA1) = UH1            |                   | 34 |
| 35 |                   | DOT   | SL1      | # VA1 . UH1                                 | B7                | 35 |
| 36 |                   |       | VACT1    |                                             |                   | 36 |
| 37 |                   | BDSU  | STADR    | #                                           | PL00D             | 37 |
| 38 |                   | STODL | DELVCSI  |                                             |                   | 38 |
| 39 |                   |       | INITST   | # 10 FPS                                    |                   | 39 |
| 40 | CSI/B1            | STORE | DELDV    |                                             |                   | 40 |
| 41 |                   | DLOAD | DAD      | # IF LOOPCT = 16                            |                   | 41 |
| 42 |                   |       | LOOPCT   |                                             |                   | 42 |
| 43 |                   |       | 1DPB28   |                                             |                   | 43 |
| 44 |                   | STORE | LOOPCT   |                                             |                   | 44 |
| 45 |                   | DSU   | AXT,2    |                                             |                   | 45 |
| 46 |                   |       | LOOPMX   |                                             |                   | 46 |
| 47 |                   |       | 6        |                                             |                   | 47 |
| 48 |                   | BPL   |          |                                             |                   | 48 |
| 49 | CSI/B2            |       | SCNDSOL  |                                             |                   | 49 |
| 50 |                   | SETPD |          |                                             |                   | 50 |
| 51 |                   |       | OD       |                                             |                   | 51 |
| 52 |                   |       |          |                                             |                   | 52 |
| 53 |                   |       |          |                                             |                   | 53 |
| 54 |                   |       |          |                                             |                   | 54 |
| 55 |                   |       |          |                                             |                   | 55 |
| 56 |                   |       |          |                                             |                   | 56 |
| 57 |                   |       |          |                                             |                   | 57 |
| 58 |                   |       |          |                                             |                   | 58 |
| 59 |                   |       |          |                                             |                   | 59 |
| 60 |                   |       |          |                                             |                   | 60 |

[illegible]

[illegible]

[illegible]

[illegible]

|    |                   |        |         |                                        |           |
|----|-------------------|--------|---------|----------------------------------------|-----------|
| 1  | # TSE TSS TTE TTS |        |         | PAGE 050                               |           |
| 2  |                   | DSQ    | TLOAD   | # TEMP1**2                             | B58       |
| 3  |                   |        | MPAC    |                                        |           |
| 4  |                   | PDVL   | DOT     | #                                      | PL11D     |
| 5  |                   |        | RACT3   |                                        |           |
| 6  |                   |        | RACT3   |                                        |           |
| 7  |                   | TLOAD  | DCOMP   | # RA3 . RA3                            |           |
| 8  |                   |        | MPAC    |                                        |           |
| 9  |                   | PDVL   | DOT     | # RP3 . RP3                            | B58 PL14D |
| 10 |                   |        | RPASS3  |                                        |           |
| 11 |                   |        | RPASS3  | #                                      | PL11D     |
| 12 |                   | TAD    | TAD     | # TEMP1**2 + RA3.RA3 + RP3.RP3 = TEMP2 | PL08D     |
| 13 |                   | BPL    | DLOAD   |                                        |           |
| 14 |                   |        | K10RK2  |                                        |           |
| 15 |                   |        | LOOPCT  |                                        |           |
| 16 |                   | DSU    | AXT,2   |                                        |           |
| 17 |                   |        | 1DPB28  |                                        |           |
| 18 |                   |        | 1D      |                                        |           |
| 19 |                   | BZE    |         |                                        |           |
| 20 |                   |        | ALMXITA |                                        |           |
| 21 |                   | DLOAD  | SR1     |                                        |           |
| 22 |                   |        | DELDV   |                                        |           |
| 23 |                   | STORE  | DELDV   |                                        |           |
| 24 |                   | BDSU   |         |                                        |           |
| 25 |                   |        | DVPREV  |                                        |           |
| 26 |                   | STCALL | DELVCSI |                                        |           |
| 27 |                   |        | CSI/B1  |                                        |           |
| 28 | K10RK2            | SQRT   | PUSH    | # TEMP3 = TEMP2**.5                    | B29 PL10D |
| 29 |                   | DCOMP  | DSU     |                                        |           |
| 30 |                   |        | 06D     | # -TEMP1-TEMP3 = K2 AT 10D             |           |
| 31 |                   | STODL  | 10D     | #                                      | PL08D     |
| 32 |                   | DSU    | STADR   | #                                      | PL06D     |
| 33 |                   | STORE  | 12D     | # -TEMP1+TEMP3 = K1 AT 12D             |           |
| 34 |                   | ABS    |         |                                        |           |
| 35 |                   | STODL  | 14D     |                                        |           |
| 36 |                   |        | 10D     |                                        |           |
| 37 |                   | ABS    | DSU     |                                        |           |
| 38 |                   |        | 14D     |                                        |           |
| 39 |                   | BMN    | DLOAD   |                                        |           |
| 40 |                   |        | K2.     |                                        |           |
| 41 |                   |        | 12D     |                                        |           |
| 42 |                   | STORE  | 10D     | # K = K1                               |           |
| 43 | K2.               | DLOAD  |         |                                        |           |
| 44 |                   |        | 10D     |                                        |           |
| 45 |                   | VXSC   | VSL1    |                                        |           |
| 46 |                   | VAD    | UNIT    | # V = RA3 + KU UNIT                    | B1        |
| 47 |                   |        | RACT3   |                                        |           |
| 48 |                   | PDVL   | UNIT    |                                        |           |
| 49 |                   |        | RPASS3  | #                                      | PL06D     |
| 50 |                   | PDVL   | UNIT    |                                        |           |
| 51 |                   |        | VPASS3  | #                                      | PL12D     |
| 52 |                   |        |         |                                        |           |
| 53 |                   |        |         |                                        |           |
| 54 |                   |        |         |                                        |           |
| 55 |                   |        |         |                                        |           |
| 56 |                   |        |         |                                        |           |
| 57 |                   |        |         |                                        |           |
| 58 |                   |        |         |                                        |           |
| 59 |                   |        |         |                                        |           |
| 60 |                   |        |         |                                        |           |

|    |                |        |          |                                       |          |       |    |
|----|----------------|--------|----------|---------------------------------------|----------|-------|----|
| 1  | # USE TYPE 113 |        |          |                                       | PAGE 659 |       | 1  |
| 2  |                | VXV    | PDVL     | # UVP3 X URP3                         |          | PL18D | 2  |
| 3  |                |        | 06D      |                                       |          |       | 3  |
| 4  |                |        | 06D      |                                       |          |       | 4  |
| 5  |                | VXV    | DOT      |                                       |          |       | 5  |
| 6  |                |        | 00D      |                                       |          |       | 6  |
| 7  |                | STADR  |          | #                                     |          | PL12D | 7  |
| 8  |                | STOVL  | 12D      | # (URP3 X V).(URP3 X URP3)=TEMP       |          | PL06D | 8  |
| 9  |                | DOT    | SL1      | #                                     |          | PL00D | 9  |
| 10 |                | ARCCOS | SIGN     |                                       |          |       | 10 |
| 11 |                |        | 12D      | #                                     | B0       |       | 11 |
| 12 |                | SR1    | PUSH     | # GAMMA = SIGN(TEMP)ARCOS(UNITV.URP3) |          | PL02D | 12 |
| 13 |                | BON    | DLOAD    |                                       |          |       | 13 |
| 14 |                |        | S32.1F2  |                                       |          |       | 14 |
| 15 |                |        | FRSTPAS  |                                       |          |       | 15 |
| 16 |                |        | 00D      | # NOT THE FIRST PASS OF A CYCLE       |          |       | 16 |
| 17 |                | DSU    | PDDL     | # GAMMA-GAMPREV                       | B1       | PL04D | 17 |
| 18 |                |        | GAMPREV  |                                       |          |       | 18 |
| 19 |                |        | DELVCSI  |                                       |          |       | 19 |
| 20 |                | DSU    | NORM     | #                                     | B7       |       | 20 |
| 21 |                |        | DVPREV   |                                       |          |       | 21 |
| 22 |                |        | X1       |                                       |          |       | 22 |
| 23 |                | BDDV   | PDDL     | # (GAM-GAMPREV)/(DV-DVPREV)           | B-6+N1   | PL06D | 23 |
| 24 |                |        | 02D      | # = SLOPE                             |          |       | 24 |
| 25 |                |        | DELVCSI  |                                       |          |       | 25 |
| 26 |                | STORE  | DVPREV   |                                       |          |       | 26 |
| 27 |                | BOFF   | BOFF     |                                       |          |       | 27 |
| 28 |                |        | S32.1F3A |                                       |          |       | 28 |
| 29 |                |        | THRDCHK  |                                       |          |       | 29 |
| 30 |                |        | S32.1F3B |                                       |          |       | 30 |
| 31 |                |        | THRDCHK  |                                       |          |       | 31 |
| 32 |                | DLOAD  | DMP      |                                       |          |       | 32 |
| 33 |                |        | 02D      |                                       |          |       | 33 |
| 34 |                |        | GAMPREV  |                                       |          |       | 34 |
| 35 |                | BPL    | DLOAD    |                                       |          |       | 35 |
| 36 |                |        | FIFTYFPS |                                       |          |       | 36 |
| 37 |                |        | INITST   |                                       |          |       | 37 |
| 38 |                | SIGN   |          |                                       |          |       | 38 |
| 39 |                |        | DELDV    |                                       |          |       | 39 |
| 40 |                | STORE  | DELDV    |                                       |          |       | 40 |
| 41 |                | SET    | CLEAR    |                                       |          |       | 41 |
| 42 |                |        | S32.1F3A |                                       |          |       | 42 |
| 43 |                |        | S32.1F3B |                                       |          |       | 43 |
| 44 | FRSTPAS        | DLOAD  |          |                                       |          |       | 44 |
| 45 |                |        | 00D      |                                       |          |       | 45 |
| 46 |                | STODL  | GAMPREV  |                                       |          |       | 46 |
| 47 |                |        | DELVCSI  |                                       |          |       | 47 |
| 48 |                | STORE  | DVPREV   |                                       |          |       | 48 |
| 49 |                | DSU    | CLEAR    |                                       |          |       | 49 |
| 50 |                |        | DELDV    |                                       |          |       | 50 |
| 51 |                |        | S32.1F2  |                                       |          |       | 51 |
| 52 |                |        |          |                                       |          |       | 52 |
| 53 |                |        |          |                                       |          |       | 53 |
| 54 |                |        |          |                                       |          |       | 54 |
| 55 |                |        |          |                                       |          |       | 55 |
| 56 |                |        |          |                                       |          |       | 56 |
| 57 |                |        |          |                                       |          |       | 57 |
| 58 |                |        |          |                                       |          |       | 58 |
| 59 |                |        |          |                                       |          |       | 59 |
| 60 |                |        |          |                                       |          |       | 60 |



|    |          |        |          |    |
|----|----------|--------|----------|----|
| 1  |          |        |          | 1  |
| 2  |          | STCALL | DELVCSI  | 2  |
| 3  |          |        | CSI/B1   | 3  |
| 4  | THRDCHK  | BON    | BON      | 4  |
| 5  |          |        | S32.1F3A | 5  |
| 6  |          |        | NEWTN    | 6  |
| 7  |          |        | S32.1F3B | 7  |
| 8  |          |        | NEWTN    | 8  |
| 9  | FIFTYFPS | DLOAD  | SIGN     | 9  |
| 10 |          |        | FIFPSDP  | 10 |
| 11 |          |        | 04D      | 11 |
| 12 |          | SIGN   |          | 12 |
| 13 |          |        | GAMPREV  | 13 |
| 14 |          | STORE  | DELDV    | 14 |
| 15 |          | DCOMP  | DAD      | 15 |
| 16 |          |        | DELVCSI  | 16 |
| 17 |          | STODL  | DELVCSI  | 17 |
| 18 |          |        | 00D      | 18 |
| 19 |          | SET    | SET      | 19 |
| 20 |          |        | S32.1F3B | 20 |
| 21 |          |        | S32.1F3A | 21 |
| 22 |          | STCALL | GAMPREV  | 22 |
| 23 |          |        | CSI/B2   | 23 |
| 24 | NEWTN    | DLOAD  | NORM     | 24 |
| 25 |          |        | 04D      | 25 |
| 26 |          |        | X2       | 26 |
| 27 |          | BDDV   | XSU,1    | 27 |
| 28 |          |        | 00D      | 28 |
| 29 |          |        | X2       | 29 |
| 30 |          | SR*    |          | 30 |
| 31 |          |        | 0,1      | 31 |
| 32 |          | STODL  | DELDV    | 32 |
| 33 |          |        | 00D      | 33 |
| 34 |          | STORE  | GAMPREV  | 34 |
| 35 |          | DLOAD  | ABS      | 35 |
| 36 |          |        | DELDV    | 36 |
| 37 |          | PUSH   | DSU      | 37 |
| 38 |          |        | EPSILN1  | 38 |
| 39 |          | BMN    | DLOAD    | 39 |
| 40 |          |        | CSI/SOL  | 40 |
| 41 |          | DSU    | BMN      | 41 |
| 42 |          |        | DELMAX1  | 42 |
| 43 |          |        | CSISTEP  | 43 |
| 44 |          | DLOAD  | SIGN     | 44 |
| 45 |          |        | DELMAX1  | 45 |
| 46 |          |        | DELDV    | 46 |
| 47 |          | STORE  | DELDV    | 47 |
| 48 | CSISTEP  | DLOAD  | DSU      | 48 |
| 49 |          |        | DELVCSI  | 49 |
| 50 |          |        | DELDV    | 50 |
| 51 |          | STCALL | DELVCSI  | 51 |
| 52 |          |        |          | 52 |
| 53 |          |        |          | 53 |
| 54 |          |        |          | 54 |
| 55 |          |        |          | 55 |
| 56 |          |        |          | 56 |
| 57 |          |        |          | 57 |
| 58 |          |        |          | 58 |
| 59 |          |        |          | 59 |
| 60 |          |        |          | 60 |

#

PL08D

| #  | CSL     | PSS    | TLE              | TTD |
|----|---------|--------|------------------|-----|
| 1  |         |        |                  |     |
| 2  |         |        |                  |     |
| 3  | CSI/SOL | DLOAD  | CSI/B1<br>AXT,2  |     |
| 4  |         |        | POSTCSI          |     |
| 5  |         | LXA,1  | 2                |     |
| 6  |         |        |                  |     |
| 7  |         | DSU*   | RTX1<br>BMN      |     |
| 8  |         |        | PMINE -2,1       |     |
| 9  |         |        |                  |     |
| 10 |         | AXT,2  | SCNDSOL<br>DLOAD |     |
| 11 |         |        | 3                |     |
| 12 |         |        |                  |     |
| 13 |         | DSU*   | POSTCDH<br>BMN   |     |
| 14 |         |        | PMINE -2,1       |     |
| 15 |         |        |                  |     |
| 16 |         | DLOAD  | SCNDSOL<br>DSU   |     |
| 17 |         |        | TCDH             |     |
| 18 |         |        |                  |     |
| 19 |         | STORE  | TCSI             |     |
| 20 |         | AXT,2  | T1TOT2           |     |
| 21 |         |        | DSU              |     |
| 22 |         |        | 4                |     |
| 23 |         | BMN    | TMIN<br>AXT,2    |     |
| 24 |         |        |                  |     |
| 25 |         |        | SCNDSOL          |     |
| 26 |         | DLOAD  | 5                |     |
| 27 |         |        | DSU              |     |
| 28 |         |        | TTPI             |     |
| 29 |         |        | TCDH             |     |
| 30 |         | STORE  | T2TOT3           |     |
| 31 |         | DSU    | BPL              |     |
| 32 |         |        | TMIN             |     |
| 33 |         |        | P32/P72C         |     |
| 34 | SCNDSOL | BON    | BOFF             |     |
| 35 |         |        | S32.1F3A         |     |
| 36 |         |        | ALMXIT           |     |
| 37 |         |        | S32.1F3B         |     |
| 38 |         |        | ALMXIT           |     |
| 39 |         | SXA,2  | DLOAD            |     |
| 40 |         |        | CSIALRM          |     |
| 41 |         |        | P30ZERO          |     |
| 42 |         | CLEAR  | SET              |     |
| 43 |         |        | S32.1F1          |     |
| 44 |         |        | S32.1F2          |     |
| 45 |         | CLEAR  | CLEAR            |     |
| 46 |         |        | S32.1F3A         |     |
| 47 |         |        | S32.1F3B         |     |
| 48 |         | STCALL | LOOPCT           |     |
| 49 |         |        | CSI/B            |     |
| 50 |         |        |                  |     |
| 51 |         |        |                  |     |
| 52 |         |        |                  |     |
| 53 |         |        |                  |     |
| 54 |         |        |                  |     |
| 55 |         |        |                  |     |
| 56 |         |        |                  |     |
| 57 |         |        |                  |     |
| 58 |         |        |                  |     |
| 59 |         |        |                  |     |
| 60 |         |        |                  |     |

# \*\*\*\*\* ADVANCE \*\*\*\*\*  
#  
# SUBROUTINES USED  
#       PRECSET  
#       ROTATE

|         |                        |                             |
|---------|------------------------|-----------------------------|
| ADVANCE | STQ                    | DLOAD<br>SUBEXIT            |
|         |                        | TIG                         |
|         | STCALL                 | TDEC1<br>PRECSET            |
|         | SET                    | VLOAD<br>XDELVFLG<br>VPASS3 |
|         | STORE<br>STOVL         | VPASS2<br>VPASS1<br>RPASS3  |
|         | STORE<br>STORE<br>UNIT | RPASS2<br>RPASS1<br>VXV     |
|         |                        | VPASS1                      |
|         | UNIT<br>STOVL          | UP1                         |
|         | STCALL                 | RACT3<br>RTIG<br>ROTATE     |
|         | STORE<br>STOVL         | RACT2<br>RACT1<br>VACT3     |
|         | STCALL                 | VTIG<br>ROTATE              |
|         | STORE<br>STCALL        | VACT2<br>VACT1<br>SUBEXIT   |



|    |                      |       |      |    |
|----|----------------------|-------|------|----|
| 1  |                      |       |      | 1  |
| 2  | # ***** ROTATE ***** |       |      | 2  |
| 3  |                      |       |      | 3  |
| 4  | ROTATE               | PUSH  | PUSH | 4  |
| 5  |                      | DOT   | VXSC | 5  |
| 6  |                      |       | UP1  | 6  |
| 7  |                      |       | UP1  | 7  |
| 8  |                      | VSL2  | BVSU | 8  |
| 9  |                      | UNIT  | PDVL | 9  |
| 10 |                      | ABVAL | VXSC | 10 |
| 11 |                      | VSL1  | RVQ  | 11 |
| 12 |                      |       |      | 12 |
| 13 |                      |       |      | 13 |
| 14 |                      |       |      | 14 |
| 15 |                      |       |      | 15 |
| 16 |                      |       |      | 16 |
| 17 |                      |       |      | 17 |
| 18 |                      |       |      | 18 |
| 19 |                      |       |      | 19 |
| 20 |                      |       |      | 20 |
| 21 |                      |       |      | 21 |
| 22 |                      |       |      | 22 |
| 23 |                      |       |      | 23 |
| 24 |                      |       |      | 24 |
| 25 |                      |       |      | 25 |
| 26 |                      |       |      | 26 |
| 27 |                      |       |      | 27 |
| 28 |                      |       |      | 28 |
| 29 |                      |       |      | 29 |
| 30 |                      |       |      | 30 |
| 31 |                      |       |      | 31 |
| 32 |                      |       |      | 32 |
| 33 |                      |       |      | 33 |
| 34 |                      |       |      | 34 |
| 35 |                      |       |      | 35 |
| 36 |                      |       |      | 36 |
| 37 |                      |       |      | 37 |
| 38 |                      |       |      | 38 |
| 39 |                      |       |      | 39 |
| 40 |                      |       |      | 40 |
| 41 |                      |       |      | 41 |
| 42 |                      |       |      | 42 |
| 43 |                      |       |      | 43 |
| 44 |                      |       |      | 44 |
| 45 |                      |       |      | 45 |
| 46 |                      |       |      | 46 |
| 47 |                      |       |      | 47 |
| 48 |                      |       |      | 48 |
| 49 |                      |       |      | 49 |
| 50 |                      |       |      | 50 |
| 51 |                      |       |      | 51 |
| 52 |                      |       |      | 52 |
| 53 |                      |       |      | 53 |
| 54 |                      |       |      | 54 |
| 55 |                      |       |      | 55 |
| 56 |                      |       |      | 56 |
| 57 |                      |       |      | 57 |
| 58 |                      |       |      | 58 |
| 59 |                      |       |      | 59 |
| 60 |                      |       |      | 60 |

```
***** INTINTNA *****
```

|          |      |                         |
|----------|------|-------------------------|
| INTINT2C | PDDL | PDDL<br>TCSI<br>TCDH    |
|          | PDDL | PUSH<br>TWOPI           |
|          | GOTO |                         |
| INTINT3P | PDDL | INTINT<br>PDDL<br>TCDH  |
|          | PDDL | TTPI<br>PUSH<br>P30ZERO |
|          | GOTO | INTINT                  |

INTINT

# \*\*\*\*\* S32/33.1 \*\*\*\*\*  
#  
# SUBROUTINES USED  
# S32/33.X

|          |       |          |
|----------|-------|----------|
| S32/33.1 | STQ   | AXT,1    |
|          |       | SUBEXIT  |
|          | VN    | 0681     |
|          | CALL  |          |
|          | CALL  | DISDVLVC |
|          |       |          |
|          | VLOAD | S32/33.X |
|          |       | VXM      |
|          |       | DELVLVC  |
|          |       | OD       |
|          | VSL1  |          |
|          | STORE | DELVSIN  |
|          | PUSH  | ABVAL    |
|          | STOVL | DELVSAB  |
|          | GOTO  |          |
|          |       | SUBEXIT  |

# \*\*\*\*\* S32/33.X \*\*\*\*\*

|          |       |       |
|----------|-------|-------|
| S32/33.X | SETPD | VLOAD |
|          |       | 6D    |
|          |       | UP1   |
|          | VCOMP | PDVL  |
|          |       | RACT1 |
|          | UNIT  | VCOMP |
|          | PUSH  | VXV   |
|          |       | UP1   |
|          | VSL1  |       |
|          | STORE | OD    |
|          | RVQ   |       |

1412THE

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60

1

# \*\*\*\*\* CDHMVR \*\*\*\*\*  
#  
# SUBROUTINES USED  
#       VECSHIFT  
#       TIMETHET  
#       SHIFTR1

|        |        |          |                          |
|--------|--------|----------|--------------------------|
| CDHMVR | STQ    | VLOAD    |                          |
|        |        | SUBEXIT  |                          |
|        |        | RACT2    |                          |
|        | PUSH   | UNIT     |                          |
|        | STOVL  | UNVEC    | # UR SUB A               |
|        |        | RPASS2   |                          |
|        | UNIT   | DOT      |                          |
|        |        | UNVEC    |                          |
|        | PUSH   | SL1      |                          |
|        | STODL  | CSTH     |                          |
|        | DSQ    | PDDL     |                          |
|        |        | DP1/4TH  |                          |
|        | SR2    | DSU      |                          |
|        | SQRT   | SL1      |                          |
|        | PDVL   | VCOMP    |                          |
|        | VXV    |          |                          |
|        |        | RPASS2   |                          |
|        | DOT    | PDDL     |                          |
|        |        | UP1      |                          |
|        | SIGN   | STADR    |                          |
|        | STOVL  | SNTH     |                          |
|        |        | RPASS2   |                          |
|        | PDVL   | CALL     |                          |
|        |        | VPASS2   |                          |
|        |        | VECSHIFT |                          |
|        | STOVL  | VVEC     |                          |
|        | CLEAR  |          |                          |
|        |        | RVSU     |                          |
|        | STCALL | RVEC     |                          |
|        |        | TIMETHET |                          |
|        | LXA,2  | VSL*     |                          |
|        |        | RTX2     |                          |
|        |        | 0,2      |                          |
|        | STORE  | 18D      |                          |
|        | DOT    | SL1R     |                          |
|        |        | UNVEC    |                          |
|        | PDVL   | ABVAL    | # OD = V SUB PV          |
|        | SL*    | PDVL     |                          |
|        |        | 0,2      |                          |
|        |        | RACT2    |                          |
|        | ABVAL  | PDDL     | # 2D = LENGTH OF R SUB A |
|        | DSU    |          |                          |



[illegible]

| # P32-P35_P72-P75 |                 |  | PAGE 649 |                                       |  |
|-------------------|-----------------|--|----------|---------------------------------------|--|
| 1                 | STCALL DELVEET2 |  |          | # DELTA VCDH -- REFERENCE COORDINATES |  |
| 2                 | SUBEXIT         |  |          |                                       |  |
| 3                 |                 |  |          |                                       |  |
| 4                 |                 |  |          |                                       |  |
| 5                 |                 |  |          |                                       |  |
| 6                 |                 |  |          |                                       |  |
| 7                 |                 |  |          |                                       |  |
| 8                 |                 |  |          |                                       |  |
| 9                 |                 |  |          |                                       |  |
| 10                |                 |  |          |                                       |  |
| 11                |                 |  |          |                                       |  |
| 12                |                 |  |          |                                       |  |
| 13                |                 |  |          |                                       |  |
| 14                |                 |  |          |                                       |  |
| 15                |                 |  |          |                                       |  |
| 16                |                 |  |          |                                       |  |
| 17                |                 |  |          |                                       |  |
| 18                |                 |  |          |                                       |  |
| 19                |                 |  |          |                                       |  |
| 20                |                 |  |          |                                       |  |
| 21                |                 |  |          |                                       |  |
| 22                |                 |  |          |                                       |  |
| 23                |                 |  |          |                                       |  |
| 24                |                 |  |          |                                       |  |
| 25                |                 |  |          |                                       |  |
| 26                |                 |  |          |                                       |  |
| 27                |                 |  |          |                                       |  |
| 28                |                 |  |          |                                       |  |
| 29                |                 |  |          |                                       |  |
| 30                |                 |  |          |                                       |  |
| 31                |                 |  |          |                                       |  |
| 32                |                 |  |          |                                       |  |
| 33                |                 |  |          |                                       |  |
| 34                |                 |  |          |                                       |  |
| 35                |                 |  |          |                                       |  |
| 36                |                 |  |          |                                       |  |
| 37                |                 |  |          |                                       |  |
| 38                |                 |  |          |                                       |  |
| 39                |                 |  |          |                                       |  |
| 40                |                 |  |          |                                       |  |
| 41                |                 |  |          |                                       |  |
| 42                |                 |  |          |                                       |  |
| 43                |                 |  |          |                                       |  |
| 44                |                 |  |          |                                       |  |
| 45                |                 |  |          |                                       |  |
| 46                |                 |  |          |                                       |  |
| 47                |                 |  |          |                                       |  |
| 48                |                 |  |          |                                       |  |
| 49                |                 |  |          |                                       |  |
| 50                |                 |  |          |                                       |  |
| 51                |                 |  |          |                                       |  |
| 52                |                 |  |          |                                       |  |
| 53                |                 |  |          |                                       |  |
| 54                |                 |  |          |                                       |  |
| 55                |                 |  |          |                                       |  |
| 56                |                 |  |          |                                       |  |
| 57                |                 |  |          |                                       |  |
| 58                |                 |  |          |                                       |  |
| 59                |                 |  |          |                                       |  |
| 60                |                 |  |          |                                       |  |



|    |                       |        |            |    |
|----|-----------------------|--------|------------|----|
| 1  |                       |        |            | 1  |
| 2  | # ***** COMPTGO ***** |        |            | 2  |
| 3  | #                     |        |            | 3  |
| 4  | # SUBROUTINES USED    |        |            | 4  |
| 5  | # CLOKTASK            |        |            | 5  |
| 6  | # 2PHSCHNG            |        |            | 6  |
| 7  |                       |        |            | 7  |
| 8  |                       | BANK   | 35         | 8  |
| 9  |                       | SETLOC | CSI/CDH    | 9  |
| 10 |                       | BANK   |            | 10 |
| 11 |                       |        |            | 11 |
| 12 |                       | EBANK= | RTRN       | 12 |
| 13 |                       |        |            | 13 |
| 14 |                       | COUNT* | \$\$/P3575 | 14 |
| 15 |                       |        |            | 15 |
| 16 | COMPTGO               | EXTEND |            | 16 |
| 17 |                       | QXCH   | RTRN       | 17 |
| 18 |                       | CAF    | ZERO       | 18 |
| 19 |                       | TS     | DISPDEX    | 19 |
| 20 |                       | CAF    | BIT2       | 20 |
| 21 |                       | INHINT |            | 21 |
| 22 |                       | TC     | WAITLIST   | 22 |
| 23 |                       | EBANK= | WHICH      | 23 |
| 24 |                       | 2CADR  | CLOKTASK   | 24 |
| 25 |                       |        |            | 25 |
| 26 |                       | TC     | 2PHSCHNG   | 26 |
| 27 |                       | OCT    | 40036      | 27 |
| 28 |                       | OCT    | 05024      | 28 |
| 29 |                       | OCT    | 13000      | 29 |
| 30 |                       | TC     | RTRN       | 30 |
| 31 |                       |        |            | 31 |
| 32 |                       |        |            | 32 |
| 33 |                       |        |            | 33 |
| 34 |                       |        |            | 34 |
| 35 |                       |        |            | 35 |
| 36 |                       |        |            | 36 |
| 37 |                       |        |            | 37 |
| 38 |                       |        |            | 38 |
| 39 |                       |        |            | 39 |
| 40 |                       |        |            | 40 |
| 41 |                       |        |            | 41 |
| 42 |                       |        |            | 42 |
| 43 |                       |        |            | 43 |
| 44 |                       |        |            | 44 |
| 45 |                       |        |            | 45 |
| 46 |                       |        |            | 46 |
| 47 |                       |        |            | 47 |
| 48 |                       |        |            | 48 |
| 49 |                       |        |            | 49 |
| 50 |                       |        |            | 50 |
| 51 |                       |        |            | 51 |
| 52 |                       |        |            | 52 |
| 53 |                       |        |            | 53 |
| 54 |                       |        |            | 54 |
| 55 |                       |        |            | 55 |
| 56 |                       |        |            | 56 |
| 57 |                       |        |            | 57 |
| 58 |                       |        |            | 58 |
| 59 |                       |        |            | 59 |
| 60 |                       |        |            | 60 |

## # GROUND\_TRACKING\_DETERMINATION\_PROGRAM

# GROUND TRACKING DETERMINATION PROGRAM P21  
# PROGRAM DESCRIPTION

# MOD NO - 1  
# MOD BY - N.M.NEVILLE  
# FUNCTIONAL DECRPTION-

#  
# TO PROVIDE THE ASTRONAUT DETAILS OF THE LM OR CSM GROUND TRACK WITHOUT  
# THE NEED FOR GROUND COMMUNICATION (REQUESTED BY DSKY).

# CALLING SEQUENCE -

#  
# ASTRONAUT REQUEST THROUGH DSKY V37E21E

# SUBROUTINES CALLED-

#  
# GOPERF4

# GOFLASH  
# THISPREC  
# OTHPREC

# LAT-LONG  
# NORMAL EXIT MODES-

#  
# ASTRONAUT REQUEST TROUGH DSKY TO TERMINATE PROGRAM V34E

# ALARM OR ABORT EXIT MODES-

#  
# NONE  
# OUTPUT -

#  
# OCTAL DISPLAY OF OPTION CODE AND VEHICLE WHOSE GROUND TRACK IS TO BE  
# COMPUTED

#       OPTION CODE       00002

#       THIS               00001

#       OTHER              00002

# DECIMAL DISPLAY OF TIME TO BE INTEGRATED TO HOURS , MINUTES , SECONDS

# DECIMAL DISPLAY OF LAT, LONG, ALT

# ERASABLE INITIALIZATION REQUIRED

#  
# AXO               2DEC     4.652459653 E-5       RADIANS       %68-69 CONSTANTS"

#  
# -AYO             2DEC     2.147535898 E-5       RADIANS

#  
# AZO               2DEC     .7753206164        REVOLUTIONS

# FOR LUNAR ORBITS 504LM VECTOR IS NEEDED

#  
# 504LM            2DEC     -2.700340600 E-5       RADIANS

#  
# 504LM \_2         2DEC     -7.514128400 E-4       RADIANS

#  
# 504LM \_4         2DEC     \_2.553198641 E-4       RADIANS

#  
# NONE  
# DEBRIS

#

```
1 # CENTRALS-A,Q,L
2 # OTHER-THOSE USED BY THE ABOVE LISTED SUBROUTINES
3 # SEE LEMPREC,LAT-LONG
4 SBANK= LOWSUPER # FOR LOW 2CADR'S.
5
6 BANK 33
7 SETLOC P20S
8 BANK
9
10 EBANK= P21TIME
11 COUNT* $$/P21
12
13 PROG21 CAF ONE
14 TS OPTION2 # ASSUMED VEHICLE IS LM , R2 = 00001
15 CAF BIT2 # OPTION 2
16
17 TC BANKCALL
18 CADR GOPERF4
19 TC GOTOP00H # TERMINATE
20
21 P21PROG1 TC +2 # PROCEED VALUE OF ASSUMED VEHICLE OK
22 TC -5 # R2 LOADED THROUGH DSKY
23 CAF V6N34 # LOAD DESIRED TIME OF LAT-LONG.
24
25 TC BANKCALL
26 CADR GOFLASH
27 TC GOTOP00H # TERM
28
29 TC +2 # PROCEED VALUES OK
30 TC -5 # TIME LOADED THROUGH DSKY
31 TC INTPRET
32
33 DLOAD
34
35 STCALL DSPTM1
36 TDEC1 # INTEGRATE TO TIME SPECIFIED IN TDEC
37
38 BON INTSTALL
39 CLEAR
40 P21FLAG
41
42 P21CONT # ON---RECYCLE USING BASE VECTOR
43 VINTFLAG # OFF---IST PASS CALL BASE VECTOR
44
45 SLOAD SR1
46
47 BHIZ OPTION2
48 SET
49 +2 # ZERO--THIS VEHICLE(LM)
50
51 CLEAR VINTFLAG # ONE--OTHER VEHICLE(CM)
52 CLEAR
53 DIMOFLAG
54
55 CALL INTYPFLG # PRECISION
56
57 INTEGRV # CALCULATE
58
59 GOTO # -AND
60 P21VSAVE # -SAVE BASE VECTOR
61
62 P21CONT VLOAD
63
64 STOVL P21BASER # RECYCLE--INTEG FROM BASE VECTOR
65 RCV # --POS
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

|    |          |       |           |                                |
|----|----------|-------|-----------|--------------------------------|
| 1  |          |       | P21BASEV  |                                |
| 2  |          | STODL | VCV       | # --VEL                        |
| 3  |          |       | P21TIME   |                                |
| 4  |          | STORE | TET       | # --TIME                       |
| 5  |          | CLEAR | CLEAR     |                                |
| 6  |          |       | DIMOFFLAG |                                |
| 7  |          |       | MOONFLAG  |                                |
| 8  |          | SLOAD | BZE       |                                |
| 9  |          |       | P21ORIG   |                                |
| 10 |          |       | +3        | # ZERO=EARTH                   |
| 11 |          | SET   |           | # ---2=MOON                    |
| 12 |          |       | MOONFLAG  |                                |
| 13 | +3       | CALL  |           |                                |
| 14 |          |       | INTEGRVS  |                                |
| 15 | P21VSAVE | DLOAD |           | # SAVE CURRENT BASEVECTOR      |
| 16 |          |       | TAT       |                                |
| 17 |          | STOVL | P21TIME   | # --TIME                       |
| 18 |          |       | RATT1     |                                |
| 19 |          | STOVL | P21BASER  | # --POS B-29 OR B-27           |
| 20 |          |       | VATT1     |                                |
| 21 |          | STORE | P21BASEV  | # --VEL B-07 OR B-05           |
| 22 |          | ABVAL | SL*       |                                |
| 23 |          |       | 0,2       |                                |
| 24 |          | STOVL | P21VEL    | # VEL/ FOR N91 DISP            |
| 25 |          |       | RATT      |                                |
| 26 |          | UNIT  | DOT       |                                |
| 27 |          |       | VATT      | # U(R).V                       |
| 28 |          | DDV   | ASIN      | # U(R).U(V)                    |
| 29 |          |       | P21VEL    |                                |
| 30 |          | STORE | P21GAM    | # SIN-1 U(R).U(V) , -90 TO &90 |
| 31 |          | SXA,2 | SLOAD     |                                |
| 32 |          |       | P21ORIG   | # 0=EARTH                      |
| 33 |          |       | OPTION2   |                                |
| 34 |          | SR1   | BHIZ      |                                |
| 35 |          |       | +3        |                                |
| 36 |          | GOTO  |           |                                |
| 37 |          |       | +4        |                                |
| 38 | +3       | BON   |           |                                |
| 39 |          |       | SURFFLAG  |                                |
| 40 |          |       | P21DSP    |                                |
| 41 | +4       | SET   |           |                                |
| 42 |          |       | P21FLAG   |                                |
| 43 | P21DSP   | CLEAR | SLOAD     | # GENERATE DISPLAY DATA        |
| 44 |          |       | LUNAFLAG  |                                |
| 45 |          |       | X2        |                                |
| 46 |          | BZE   | SET       |                                |
| 47 |          |       | +2        | # 0 = EARTH                    |
| 48 |          |       | LUNAFLAG  |                                |
| 49 |          | VLOAD |           |                                |
| 50 |          |       | RATT      |                                |
| 51 |          |       |           |                                |
| 52 |          |       |           |                                |
| 53 |          |       |           |                                |
| 54 |          |       |           |                                |
| 55 |          |       |           |                                |
| 56 |          |       |           |                                |
| 57 |          |       |           |                                |
| 58 |          |       |           |                                |
| 59 |          |       |           |                                |
| 60 |          |       |           |                                |

|    |        |       |                           |    |
|----|--------|-------|---------------------------|----|
| 1  |        |       |                           | 1  |
| 2  |        | STODL | ALPHAV                    | 2  |
| 3  |        |       | TAT                       | 3  |
| 4  |        | CLEAR | CALL                      | 4  |
| 5  |        |       | ERADFLAG                  | 5  |
| 6  |        |       | LAT-LONG                  | 6  |
| 7  |        | DMP   | # MPAC = ALT, METERS B-29 | 7  |
| 8  |        |       | K.01                      | 8  |
| 9  |        | STORE | P21ALT                    | 9  |
| 10 |        |       | # ALT/100 FOR N91 DISP    | 10 |
| 11 |        | EXIT  |                           | 11 |
| 12 |        | CAF   | V06N43                    | 12 |
| 13 |        | TC    | BANKCALL                  | 13 |
| 14 |        | CADR  | GOFLASH                   | 14 |
| 15 |        | TC    | GOTOPQOH                  | 15 |
| 16 |        | TC    | GOTOPQOH                  | 16 |
| 17 |        | TC    | INTPRET                   | 17 |
| 18 |        | DLOAD | DAD                       | 18 |
| 19 |        |       | P21TIME                   | 19 |
| 20 |        |       | 600SEC                    | 20 |
| 21 |        | STORE | DSPTM1                    | 21 |
| 22 |        | RTB   |                           | 22 |
| 23 | 600SEC | 2DEC  | P21PROG1                  | 23 |
| 24 |        |       | 60000                     | 24 |
| 25 |        |       | # 10 MIN                  | 25 |
| 26 | V06N43 | VN    | 00643                     | 26 |
| 27 | V6N34  | VN    | 00634                     | 27 |
| 28 | K.01   | 2DEC  | .01                       | 28 |
| 29 |        |       |                           | 29 |
| 30 |        |       |                           | 30 |
| 31 |        |       |                           | 31 |
| 32 |        |       |                           | 32 |
| 33 |        |       |                           | 33 |
| 34 |        |       |                           | 34 |
| 35 |        |       |                           | 35 |
| 36 |        |       |                           | 36 |
| 37 |        |       |                           | 37 |
| 38 |        |       |                           | 38 |
| 39 |        |       |                           | 39 |
| 40 |        |       |                           | 40 |
| 41 |        |       |                           | 41 |
| 42 |        |       |                           | 42 |
| 43 |        |       |                           | 43 |
| 44 |        |       |                           | 44 |
| 45 |        |       |                           | 45 |
| 46 |        |       |                           | 46 |
| 47 |        |       |                           | 47 |
| 48 |        |       |                           | 48 |
| 49 |        |       |                           | 49 |
| 50 |        |       |                           | 50 |
| 51 |        |       |                           | 51 |
| 52 |        |       |                           | 52 |
| 53 |        |       |                           | 53 |
| 54 |        |       |                           | 54 |
| 55 |        |       |                           | 55 |
| 56 |        |       |                           | 56 |
| 57 |        |       |                           | 57 |
| 58 |        |       |                           | 58 |
| 59 |        |       |                           | 59 |
| 60 |        |       |                           | 60 |

## # TRANSFER PHASE INITIATION (TPI) PROGRAMS (P34 AND P74)

# MOD NO -1 LOG SECTION -- P32-P35, P72-P75  
# MOD BY WHITE, P. DATE: 1 JUNE 67  
#

## # PURPOSE

# (1) TO CALCULATE THE REQUIRED DELTA V AND OTHER INITIAL CONDITIONS  
# REQUIRED BY THE ACTIVE VEHICLE FOR EXECUTION OF THE TRANSFER  
# PHASE INITIATION (TPI) MANEUVER, GIVEN --

# (A) TIME OF IGNITION TIG (TPI) OR THE ELEVATION ANGLE (E) OF  
# THE ACTIVE/PASSIVE VEHICLE LOS AT TIG (TPI).

# (B) CENTRAL ANGLE OF TRANSFER (CENTANG) FROM TIG (TPI) TO  
# INTERCEPT TIME (TIG (TPF)).

# (2) TO CALCULATE TIG (TPI) GIVEN E OR E GIVEN TIG (TPI).

# (3) TO CALCULATE THESE PARAMETERS BASED UPON MANEUVER DATA  
# APPROVED AND KEYED INTO THE DSKY BY THE ASTRONAUT.

# (4) TO DISPLAY TO THE ASTRONAUT AND THE GROUND CERTAIN DEPENDENT  
# VARIABLES ASSOCIATED WITH THE MANEUVER FOR APPROVAL BY THE  
# ASTRONAUT/GROUND.

# (5) TO STORE THE TPI TARGET PARAMETERS FOR USE BY THE DESIRED  
# THRUSTING PROGRAM.  
#

## # ASSUMPTIONS

# (1) LM ONLY -- THIS PROGRAM IS BASED UPON PREVIOUS COMPLETION OF  
# THE CONSTANT DELTA ALTITUDE (CDH) PROGRAM (P33/P73).  
# THEREFORE --

# (A) AT A SELECTED TPI TIME (NOW IN STORAGE) THE LINE OF SIGHT  
# BETWEEN THE ACTIVE AND PASSIVE VEHICLES WAS SELECTED TO BE  
# A PRESCRIBED ANGLE (E) (NOW IN STORAGE) FROM THE  
# HORIZONTAL PLANE DEFINED BY THE ACTIVE VEHICLE POSITION.

# (B) THE TIME BETWEEN CDH IGNITION AND TPI IGNITION WAS  
# COMPUTED TO BE GREATER THAN 10 MINUTES.

# (C) THE VARIATION OF THE ALTITUDE DIFFERENCE BETWEEN THE  
# ORBITS WAS MINIMIZED.

# (D) THE PERICENTER ALTITUDES OF ORBITS FOLLOWING CSI AND  
# CDH WERE COMPUTED TO BE GREATER THAN 35,000 FT FOR LUNAR



# ORBIT OR 85 NM FOR EARTH ORBIT.

# (E) THE CSI AND CDH MANEUVERS WERE ASSUMED TO BE PARALLEL TO  
# THE PLANE OF THE PASSIVE VEHICLE ORBIT. HOWEVER, CREW  
# MODIFICATION OF DELTA V (LV) COMPONENTS MAY HAVE RESULTED  
# IN AN OUT-OF-PLANE MANEUVER.

# (2) STATE VECTOR UPDATED BY P27 ARE DISALLOWED DURING AUTOMATIC  
# STATE VECTOR UPDATING INITIATED BY P20 (SEE ASSUMPTION (4)).

# (3) THIS PROGRAM MUST BE DONE OVER A TRACKING STATION FOR REAL  
# TIME GROUND PARTICIPATION IN DATA INPUT AND OUTPUT. COMPUTED  
# VARIABLES MAY BE STORED FOR LATER VERIFICATION BY THE GROUND.  
# THESE STORAGE CAPABILITIES ARE LIMITED ONLY TO THE PARAMETERS  
# FOR ONE THRUSTING MANEUVER AT A TIME EXCEPT FOR CONCENTRIC  
# FLIGHT PLAN MANEUVER SEQUENCES.

# (4) THE RENDEZVOUS RADAR MAY OR MAY NOT BE USED TO UPDATE THE LM  
# OR CSM STATE VECTORS FOR THIS PROGRAM. IF RADAR USE IS  
# DESIRED THE RADAR WAS TURNED ON AND LOCKED ON THE CSM BY  
# PREVIOUS SELECTION OF P20. RADAR SIGHTING MARKS WILL BE MADE  
# AUTOMATICALLY APPROXIMATELY ONCE A MINUTE WHEN ENABLED BY THE  
# TRACK AND UPDATE FLAGS (SEE P20). THE RENDEZVOUS TRACKING  
# MARK COUNTER IS ZEROED BY THE SELECTION OF P20 AND AFTER EACH  
# THRUSTING MANEUVER.

# (5) THE ISS NEED NOT BE ON TO COMPLETE THIS PROGRAM.

# (6) THE OPERATION OF THE PROGRAM UTILIZES THE FOLLOWING FLAGS --

# ACTIVE VEHICLE FLAG -- DESIGNATES THE VEHICLE WHICH IS  
# DOING RENDEZVOUS THRUSTING MANEUVERS TO THE PROGRAM WHICH  
# CALCULATES THE MANEUVER PARAMETERS. SET AT THE START OF  
# EACH RENDEZVOUS PRE-THRUSTING PROGRAM.

# FINAL FLAG -- SELECTS FINAL PROGRAM DISPLAYS AFTER CREW HAS  
# SELECTED THE FINAL MANEUVER COMPUTATION CYCLE.

# EXTERNAL DELTA V FLAG -- DESIGNATES THE TYPE OF STEERING  
# REQUIRED FOR EXECUTION OF THIS MANEUVER BY THE THRUSTING  
# PROGRAM SELECTED AFTER COMPLETION OF THIS PROGRAM.

# (7) ONCE THE PARAMETERS REQUIRED FOR COMPUTATION OF THE MANEUVER  
# HAVE BEEN COMPLETELY SPECIFIED, THE VALUE OF THE ACTIVE  
# VEHICLE CENTRAL ANGLE OF TRANSFER IS COMPUTED AND STURED.  
# THIS NUMBER WILL BE AVAILABLE FOR DISPLAY TO THE ASTRONAUT  
# THROUGH THE USE OF V06N52.

# THE ASTRONAUT WILL CALL THIS DISPLAY TO VERIFY THAT THE  
# CENTRAL ANGLE OF TRANSFER OF THE ACTIVE VEHICLE IS NOT WITHIN

```
1 #
2 # 170 TO 190 DEGREES. IF THE ANGLE IS WITHIN THIS ZONE THE
3 # ASTRONAUT SHOULD REASSES THE INPUT TARGETING PARAMETERS BASED
4 # UPON DELTA V AND EXPECTED MANEUVER TIME.
5 #
6 # (8) THIS PROGRAM IS SELECTED BY THE ASTRONAUT BY DSKY ENTRY --
7 #
8 # P34 IF THIS VEHICLE IS ACTIVE VEHICLE.
9 #
10 # P74 IF THIS VEHICLE IS PASSIVE VEHICLE.
11 #
12 # INPUT
13 #
14 # (1) TTPI TIME OF THE TPI MANEUVER.
15 # (2) ELEV DESIRED LOS ANGLE AT TPI
16 # (3) CENTANG ORBITAL CENTRAL ANGLE OF THE PASSIVE VEHICLE DURING
17 # TRANSFER FROM TPI TO TIME OF INTERCEPT
18 #
19 # OUTPUT
20 #
21 # (1) TRKMKCNT NUMBER OF MARKS
22 # (2) TTOGO TIME TO GO
23 # (3) +MGA MIDDLE GIMBAL ANGLE
24 # (4) TTPI COMPUTED TIME OF TPI MANEUVER
25 # OR
26 # ELEV COMPUTED LOS ANGLE AT TPI
27 # (5) POSTTPI PERIGEE ALTITUDE AFTER THE TPI MANEUVER
28 # (6) DELVTPI MAGNITUDE OF DELTA V AT TPI
29 # (7) DELVTPF MAGNITUDE OF DELTA V AT INTERCEPT
30 # (8) DVLOS DELTA VELOCITY AT TPI -- LINE OF SIGHT
31 # (9) DELVLVC DELTA VELOCITY AT TPI -- LOCAL VERTICAL COORDINATES
32 #
33 # DOWNLINK
34 #
35 # (1) TTPI TIME OF TPI MANEUVER
36 # (2) TIG TIME OF TPI MANEUVER
37 # (3) ELEV DESIRED LOS ANGLE AT TPI
38 # (4) CENTANG ORBITAL CENTRAL ANGLE OF THE PASSIVE VEHICLE DURING
39 # TRANSFER FROM TPI TO TIME OF INTERCEPT
40 # (5) DELVEET3 DELTA VELOCITY AT TPI -- REFERENCE COORDINATES
41 # (6) TPASS4 TIME OF INTERCEPT
42 #
43 # COMMUNICATION TO THRUSTING PROGRAMS
44 #
45 # (1) TIG TIME OF THE TPI MANEUVER
46 # (2) RTARG OFFSET TARGET POSITION
47 # (3) TPASS4 TIME OF INTERCEPT
48 # (4) XDELVFLG RESET TO INDICATE LAMBERT (AIMPOINT) VG COMPUTATION
49 #
50 # SUBROUTINES USED
51 #
52 # AVFLAGA
```

|    |                   |          |            |                          |    |
|----|-------------------|----------|------------|--------------------------|----|
| 1  | # P34/35.1 P74/75 |          |            |                          | 1  |
| 2  | #                 | AVFLAGP  |            |                          | 2  |
| 3  | #                 | VNPOOH   |            |                          | 3  |
| 4  | #                 | DISPLAYE |            |                          | 4  |
| 5  | #                 | SELECTMU |            |                          | 5  |
| 6  | #                 | PRECSET  |            |                          | 6  |
| 7  | #                 | S33/34.1 |            |                          | 7  |
| 8  | #                 | ALARM    |            |                          | 8  |
| 9  | #                 | BANKCALL |            |                          | 9  |
| 10 | #                 | GOFLASH  |            |                          | 10 |
| 11 | #                 | GOTOPOOH |            |                          | 11 |
| 12 | #                 | TIMETHET |            |                          | 12 |
| 13 | #                 | S34/35.2 |            |                          | 13 |
| 14 | #                 | PERIAP01 |            |                          | 14 |
| 15 | #                 | SHIFTR1  |            |                          | 15 |
| 16 | #                 | S34/35.5 |            |                          | 16 |
| 17 | #                 | VN1645   |            |                          | 17 |
| 18 |                   |          |            |                          | 18 |
| 19 |                   | SETLOC   | CSI/CDH    |                          | 19 |
| 20 |                   | BANK     |            |                          | 20 |
| 21 |                   | EBANK=   | SUBEXIT    |                          | 21 |
| 22 |                   | COUNT*   | \$\$/P3474 |                          | 22 |
| 23 | P34               | TC       | AVFLAGA    |                          | 23 |
| 24 |                   | TC       | P34/P74A   |                          | 24 |
| 25 | P74               | TC       | AVFLAGP    |                          | 25 |
| 26 | P34/P74A          | TC       | P20FLGON   | # SET UPDATFLG, TRACKFLG | 26 |
| 27 |                   | CAF      | V06N37     | # TTPI                   | 27 |
| 28 |                   | TC       | VNPOOH     |                          | 28 |
| 29 |                   | EXTEND   |            |                          | 29 |
| 30 |                   | DCA      | 130DEG     |                          | 30 |
| 31 |                   | DXCH     | CENTANG    |                          | 31 |
| 32 |                   | CAF      | P30ZERO    |                          | 32 |
| 33 |                   | TS       | NN         |                          | 33 |
| 34 |                   | TC       | DISPLAYE   | # ELEV AND CENTANG       | 34 |
| 35 |                   | TC       | INTPRET    |                          | 35 |
| 36 |                   | CLEAR    | DLOAD      |                          | 36 |
| 37 |                   |          | ETPIFLAG   |                          | 37 |
| 38 |                   |          | TTPI       |                          | 38 |
| 39 |                   | STODL    | TIG        |                          | 39 |
| 40 |                   |          | ELEV       |                          | 40 |
| 41 |                   | BZE      | SET        |                          | 41 |
| 42 |                   |          | P34/P74B   |                          | 42 |
| 43 | P34/P74B          | CALL     | ETPIFLAG   |                          | 43 |
| 44 |                   |          |            |                          | 44 |
| 45 |                   |          | SELECTMU   |                          | 45 |
| 46 | DELELO            | EQUALS   | 26D        |                          | 46 |
| 47 | P34/P74C          | DLOAD    | SET        |                          | 47 |
| 48 |                   |          | ZEROVECS   |                          | 48 |
| 49 |                   |          | ITSWICH    |                          | 49 |
| 50 |                   | BON      | CLEAR      |                          | 50 |
| 51 |                   |          | ETPIFLAG   |                          | 51 |
| 52 |                   |          |            |                          | 52 |
| 53 |                   |          |            |                          | 53 |
| 54 |                   |          |            |                          | 54 |
| 55 |                   |          |            |                          | 55 |
| 56 |                   |          |            |                          | 56 |
| 57 |                   |          |            |                          | 57 |
| 58 |                   |          |            |                          | 58 |
| 59 |                   |          |            |                          | 59 |
| 60 |                   |          |            |                          | 60 |

|    |                    |        |           |                |    |
|----|--------------------|--------|-----------|----------------|----|
| 1  |                    |        |           | 1              |    |
| 2  |                    |        |           | 2              |    |
| 3  |                    |        |           | 3              |    |
| 4  | SWCHSET<br>INTLOOP | STORE  | SWCHSET   | 4              |    |
| 5  |                    | DLOAD  | ITSWICH   | 5              |    |
| 6  |                    |        | NOMTPI    | 6              |    |
| 7  |                    |        | DAD       | 7              |    |
| 8  |                    |        | TTPI      | 8              |    |
| 9  |                    | STCALL | NOMTPI    | 9              |    |
| 10 |                    |        | TDEC1     | 10             |    |
| 11 |                    |        | PRECSET   | 11             |    |
| 12 |                    |        |           | 12             |    |
| 13 |                    |        | CALL      |                | 13 |
| 14 |                    |        | S33/34.1  | 14             |    |
| 15 |                    | BZE    | EXIT      | 15             |    |
| 16 |                    |        | SWCHCLR   | 16             |    |
| 17 |                    | TC     | ALARM     | 17             |    |
| 18 |                    | OCT    | 611       | 18             |    |
| 19 |                    | CAF    | V05N09    | 19             |    |
| 20 |                    | TC     | BANKCALL  | 20             |    |
| 21 |                    | CADR   | GOFLASH   | 21             |    |
| 22 |                    | TC     | GOTOP00H  | 22             |    |
| 23 |                    | TC     | P34/P74A  | 23             |    |
| 24 |                    | TC     | -7        | 24             |    |
| 25 |                    |        | # PROCEED | 25             |    |
| 26 |                    |        | # V32     | 26             |    |
| 27 |                    |        |           | 27             |    |
| 28 | SWCHCLR            | BONCLR | BON       | 28             |    |
| 29 |                    |        | ITSWICH   | 29             |    |
| 30 |                    |        | INTLOOP   | 30             |    |
| 31 |                    |        | ETPIFLAG  | 31             |    |
| 32 |                    |        | P34/P74D  | 32             |    |
| 33 |                    |        |           | # DISPLAY TTPI | 33 |
| 34 |                    |        | EXIT      |                | 34 |
| 35 |                    |        | TC        | DISPLA         | 35 |
| 36 |                    |        | TC        | P34/P74E       | 36 |
| 37 |                    |        |           |                | 37 |
| 38 | P34/P74D           | EXIT   |           | 38             |    |
| 39 |                    | CAF    | V06N37    | 39             |    |
| 40 |                    | TC     | VNPOOH    | 40             |    |
| 41 |                    |        |           | 41             |    |
| 42 | P34/P74E           | TC     | INTPRET   | 42             |    |
| 43 |                    | SETPD  | DLOAD     | 43             |    |
| 44 |                    |        | OD        | 44             |    |
| 45 |                    |        | RTX1      | 45             |    |
| 46 |                    | STODL  | X1        | 46             |    |
| 47 |                    |        | CENTANG   | 47             |    |
| 48 |                    |        |           | 48             |    |
| 49 |                    | PUSH   | COS       | 49             |    |
| 50 |                    | STODL  | CSTH      | 50             |    |
| 51 |                    | SIN    |           | 51             |    |
| 52 |                    | STOVL  | SNTH      | 52             |    |
| 53 |                    |        | RPASS3    | 53             |    |
| 54 |                    | VSR*   |           | 54             |    |
| 55 |                    |        |           | 55             |    |
| 56 |                    | STOVL  | 0,2       | 56             |    |
| 57 |                    |        | RVEC      | 57             |    |
| 58 |                    |        | VPASS3    | 58             |    |
| 59 |                    | VSR*   | SET       | 59             |    |
| 60 |                    |        | 0,2       | 60             |    |
| 61 |                    |        | RVSW      | 61             |    |
| 62 |                    |        |           | 62             |    |
| 63 |                    |        |           | 63             |    |
| 64 |                    |        |           | 64             |    |
| 65 |                    |        |           | 65             |    |
| 66 |                    |        |           | 66             |    |
| 67 |                    |        |           | 67             |    |
| 68 |                    |        |           | 68             |    |
| 69 |                    |        |           | 69             |    |
| 70 |                    |        |           | 70             |    |
| 71 |                    |        |           | 71             |    |
| 72 |                    |        |           | 72             |    |
| 73 |                    |        |           | 73             |    |
| 74 |                    |        |           | 74             |    |
| 75 |                    |        |           | 75             |    |
| 76 |                    |        |           | 76             |    |
| 77 |                    |        |           | 77             |    |
| 78 |                    |        |           | 78             |    |
| 79 |                    |        |           | 79             |    |
| 80 |                    |        |           | 80             |    |

[illegible]

## # RENDEZVOUS MID-COURSE MANEUVER PROGRAMS (P35 AND P75)

# MOD NO -1 LOG SECTION -- P32-P35, P72-P75  
# MOD BY WHITE, P. DATE: 1 JUNE 67  
#

## # PURPOSE

# (1) TO CALCULATE THE REQUIRED DELTA V AND OTHER INITIAL CONDITIONS  
# REQUIRED BY THE ACTIVE VEHICLE FOR EXECUTION OF THE NEXT  
# MID-COURSE CORRECTION OF THE TRANSFER PHASE OF AN ACTIVE  
# VEHICLE RENDEZVOUS.

# (2) TO DISPLAY TO THE ASTRONAUT AND THE GROUND CERTAIN DEPENDENT  
# VARIABLES ASSOCIATED WITH THE MANEUVER FOR APPROVAL BY THE  
# ASTRONAUT/GROUND.

# (3) TO STORE THE TPM TARGET PARAMETERS FOR USE BY THE DESIRED  
# THRUSTING PROGRAM.

## # ASSUMPTIONS

# (1) THE ISS NEED NOT BE ON TO COMPLETE THIS PROGRAM.

# (2) STATE VECTOR UPDATES BY P27 ARE DISALLOWED DURING AUTOMATIC  
# STATE VECTOR UPDATING INITIATED BY P20 (SEE ASSUMPTION (3)).

# (3) THE RENDEZVOUS RADAR IS ON AND IS LOCKED ON THE CSM. THIS WAS  
# DONE DURING PREVIOUS SELECTION OF P20. RADAR SIGHTING MARKS  
# WILL BE MADE AUTOMATICALLY APPROXIMATELY ONCE A MINUTE WHEN  
# ENABLED BY THE TRACK AND UPDATE FLAGS (SEE P20). THE  
# RENDEZVOUS TRACKING MARK COUNTER IS ZEROED BY THE SELECTION OF  
# P20 AND AFTER EACH THRUSTING MANEUVER.

# (4) THE OPERATION OF THE PROGRAM UTILIZES THE FOLLOWING FLAGS --

# THE ACTIVE VEHICLE FLAG -- DESIGNATES THE VEHICLE WHICH IS  
# DOING RENDEZVOUS THRUSTING MANEUVERS TO THE PROGRAM WHICH  
# CALCULATES THE MANEUVER PARAMETERS. SET AT THE START OF  
# EACH RENDEZVOUS PRE-THRUSTING PROGRAM.

# FINAL FLAG -- SELECTS FINAL PROGRAM DISPLAYS AFTER CREW HAS  
# SELECTED THE FINAL MANEUVER COMPUTATION CYCLE.

# EXTERNAL DELTA V FLAG -- DESIGNATES THE TYPE OF STEERING  
# REQUIRED FOR EXECUTION OF THIS MANEUVER BY THE THRUSTING  
# PROGRAM SELECTED AFTER COMPLETION OF THIS PROGRAM.

# (5) THE TIME OF INTERCEPT (T(INT)) WAS DEFINED BY PREVIOUS  
# COMPLETION OF THE TRANSFER PHASE INITIATION (TPI) PROGRAM  
# (P34/P74) AND IS PRESENTLY AVAILABLE IN STORAGE.  
#

```
1 #
2 # (6) ONCE THE PARAMETERS REQUIRED FOR COMPUTATION OF THE MANEUVER
3 # HAVE BEEN COMPLETELY SPECIFIED, THE VALUE OF THE ACTIVE
4 # VEHICLE CENTRAL ANGLE OF TRANSFER IS COMPUTED AND STORED.
5 # THIS NUMBER WILL BE AVAILABLE FOR DISPLAY TO THE ASTRONAUT
6 # THROUGH THE USE OF V06N52
7 #
8 # THE ASTRONAUT WILL CALL THIS DISPLAY TO VERIFY THAT THE
9 # CENTRAL ANGLE OF TRANSFER OF THE ACTIVE VEHICLE IS NOT WITHIN
10 # 170 TO 190 DEGREES. IF THE ANGLE IS WITHIN THIS ZONE THE
11 # ASTRONAUT SHOULD REASSESS THE INPUT TARGETING PARAMETERS BASED
12 # UPON DELTA V AND EXPECTED MANEUVER TIME.
13 #
14 # (7) THIS PROGRAM IS SELECTED BY THE ASTRONAUT BY DSKY ENTRY --
15 #
16 # P35 IF THIS VEHICLE IS ACTIVE VEHICLE.
17 #
18 # P75 IF THIS VEHICLE IS PASSIVE VEHICLE.
19 #
20 # INPUT
21 #
22 # (1) TPASS4 TIME OF INTERCEPT -- SAVED FROM P34/P74
23 #
24 # OUTPUT
25 #
26 # (1) TRKMKCNT NUMBER OF MARKS
27 # (2) TTOGO TIME TO GO
28 # (3) +MGA MIDOLF GIMBAL ANGLE
29 # (4) DVLOS DELTA VELOCITY AT MID -- LINE OF SIGHT
30 # (5) DELVLVC DELTA VELOCITY AT MID -- LOCAL VERTICAL COORDINATES
31 #
32 # DOWNLINK
33 #
34 # (1) TIG TIME OF THE TPM MANEUVER
35 # (2) DELVEET3 DELTA VELOCITY AT TPM -- REFERENCE COORDINATES
36 # (3) TPASS4 TIME OF INTERCEPT
37 #
38 # COMMUNICATION TO THRUSTING PROGRAMS
39 #
40 # (1) TIG TIME OF THE TPM MANEUVER
41 # (2) RTARG OFFSET TARGET POSITION
42 # (3) TPASS4 TIME OF INTERCEPT
43 # (4) XDELVFLG RESET TO INDICATE LAMBERT (AIMPOINT) VG COMPUTATION.
44 #
45 # SUBROUTINES USED
46 #
47 # AVFLAGA
48 # AVFLAGP
49 # LOADTIME
50 # SELECTMU
51 # PRECSET
52 # S34/35.1
53 # S34/35.2
```

# P34-35\_P74-75

|    |          |          |                                  |    |
|----|----------|----------|----------------------------------|----|
| 1  |          |          |                                  | 1  |
| 2  | #        | S34/35.5 |                                  | 2  |
| 3  | #        | VN1645   |                                  | 3  |
| 4  |          |          |                                  | 4  |
| 5  |          | COUNT*   | \$\$/P3575                       | 5  |
| 6  |          | EBANK=   | KT                               | 6  |
| 7  |          |          |                                  | 7  |
| 8  | P35      | TC       | AVFLAGA                          | 8  |
| 9  |          | EXTEND   |                                  | 9  |
| 10 |          |          |                                  | 10 |
| 11 |          | DCA      | ATIGINC                          | 11 |
| 12 | P75      | TC       | P35/P75A                         | 12 |
| 13 |          | TC       | AVFLAGP                          | 13 |
| 14 |          |          |                                  | 14 |
| 15 | P35/P75A | EXTEND   | PTIGINC                          | 15 |
| 16 |          | DCA      | KT                               | 16 |
| 17 |          | DXCH     |                                  | 17 |
| 18 |          | TC       | P20FLGON                         | 18 |
| 19 |          | TC       | # SET UPDATFLG, TRACKFLG         | 19 |
| 20 |          | CALL     | INTPRET                          | 20 |
| 21 |          |          |                                  | 21 |
| 22 | P35/P75B | RTB      | SELECTMU                         | 22 |
| 23 |          | LOADTIME |                                  | 23 |
| 24 |          | TSTRT    |                                  | 24 |
| 25 |          | DAD      |                                  | 25 |
| 26 |          | KT       |                                  | 26 |
| 27 |          | STORE    | TIG                              | 27 |
| 28 |          | STORE    | INTIME                           | 28 |
| 29 |          | STCALL   | TDEC1                            | 29 |
| 30 |          |          | PRECSET                          | 30 |
| 31 |          | CALL     | # ADVANCE BOTH VEHICLES          | 31 |
| 32 |          |          | S34/35.1                         | 32 |
| 33 |          | CALL     | # GET NORM AND LOS FOR TRANSFORM | 33 |
| 34 |          |          | S34/35.2                         | 34 |
| 35 |          | CALL     | # GET DELTA V(LV)                | 35 |
| 36 |          |          | S34/35.5                         | 36 |
| 37 |          | CALL     |                                  | 37 |
| 38 |          |          | VN1645                           | 38 |
| 39 |          | GOTO     |                                  | 39 |
| 40 |          |          | P35/P75B                         | 40 |
| 41 |          |          |                                  | 41 |
| 42 |          |          |                                  | 42 |
| 43 |          |          |                                  | 43 |
| 44 |          |          |                                  | 44 |
| 45 |          |          |                                  | 45 |
| 46 |          |          |                                  | 46 |
| 47 |          |          |                                  | 47 |
| 48 |          |          |                                  | 48 |
| 49 |          |          |                                  | 49 |
| 50 |          |          |                                  | 50 |
| 51 |          |          |                                  | 51 |
| 52 |          |          |                                  | 52 |
| 53 |          |          |                                  | 53 |
| 54 |          |          |                                  | 54 |
| 55 |          |          |                                  | 55 |
| 56 |          |          |                                  | 56 |
| 57 |          |          |                                  | 57 |
| 58 |          |          |                                  | 58 |
| 59 |          |          |                                  | 59 |
| 60 |          |          |                                  | 60 |



# \*\*\*\*\* S33/34.1 \*\*\*\*\*

S33/34.1

STQ

SSP  
NORMEX  
TITER

OCT  
DLOAD

40000  
SETPD  
MAX250

STOVL

OD  
SECMAX  
RACT3

STOVL

RAPREC  
VACT3

STOVL

VAPREC

STOVL

RPASS3  
RPPREC  
VPASS3

ELCALC

STORE  
CALL

VPPREC

S34/35.1 # NORMAL AND LOS

VXV

PDVL  
RACT3

# (RA\*VA)\*RA OD

PDVL

UNIT

# ULOS AT 6D

PDVL

RACT3  
VPROJ

# XCHNJ AND UP

VSL2

BVSU

UNIT

ULOS  
PDVL

# UP AT OD

DOT

PDVL

# UP.UN\*RA AT OD

DOT

OD  
SIGN

# UP IN MPAC

ULOS

SL1

ACOS

PDVL

DOT

# EA AT OD

ULOS

BPL

RACT3  
DLOAD  
TESTY

TESTY

DSU  
BOFF

DPPOS MAX  
PUSH  
DLOAD

ITSWICH

ELEX

DELEL

STODL

DELELO

DSU

ELEV

STORE

DELEL

ABS

DSU

ELEPS

|    |         |       |        |                           |    |
|----|---------|-------|--------|---------------------------|----|
| 1  |         |       |        |                           | 1  |
| 2  |         | BMN   |        |                           | 2  |
| 3  |         |       | TIMEX  | # COMMERCIALS EVERYWHERE  | 3  |
| 4  | FIGTIME | SLOAD | SR1    |                           | 4  |
| 5  |         |       | TITER  |                           | 5  |
| 6  |         | BHIZ  | LXA,1  |                           | 6  |
| 7  |         |       | NORMEX | # TOO MANY ITERATIONS     | 7  |
| 8  |         |       | MPAC   |                           | 8  |
| 9  |         | SXA,1 | VLOAD  |                           | 9  |
| 10 |         |       | TITER  |                           | 10 |
| 11 |         |       | RPASS3 |                           | 11 |
| 12 |         | UNIT  | PDDL   |                           | 12 |
| 13 |         |       | 36D    |                           | 13 |
| 14 |         | PDVL  | UNIT   |                           | 14 |
| 15 |         |       | RACT3  |                           | 15 |
| 16 |         | PDDL  |        |                           | 16 |
| 17 |         | PDDL  | PUSH   |                           | 17 |
| 18 |         |       | 36D    |                           | 18 |
| 19 |         | BDSU  |        |                           | 19 |
| 20 |         |       | 12D    |                           | 20 |
| 21 |         | STODL | 30D    | # RP - RA MAGNITUDES      | 21 |
| 22 |         |       | DPHALF |                           | 22 |
| 23 |         | DSU   | PUSH   |                           | 23 |
| 24 |         |       | ELEV   |                           | 24 |
| 25 |         | SIGN  | BMN    |                           | 25 |
| 26 |         |       | 30D    |                           | 26 |
| 27 |         |       | NORMEX |                           | 27 |
| 28 |         | DLOAD | COS    |                           | 28 |
| 29 |         | DMP   | DDV    |                           | 29 |
| 30 |         |       | 14D    |                           | 30 |
| 31 |         |       | 12D    |                           | 31 |
| 32 |         | DCOMP |        | # SINCE COS(180-A)=-COS A | 32 |
| 33 |         | STORE | 28D    |                           | 33 |
| 34 |         | ABS   | BDSU   |                           | 34 |
| 35 |         |       | DPHALF |                           | 35 |
| 36 |         | BMN   | VLOAD  |                           | 36 |
| 37 |         |       | NORMEX |                           | 37 |
| 38 |         |       | UNRM   |                           | 38 |
| 39 |         | VXV   | UNIT   |                           | 39 |
| 40 |         |       | 6D     | # UN*RA                   | 40 |
| 41 |         | DOT   | DMP    |                           | 41 |
| 42 |         |       | VACT3  |                           | 42 |
| 43 |         |       | 12D    |                           | 43 |
| 44 |         | PDVL  | VXV    |                           | 44 |
| 45 |         |       | 0D     |                           | 45 |
| 46 |         |       | VPASS3 |                           | 46 |
| 47 |         | VXV   | UNIT   |                           | 47 |
| 48 |         |       | 0D     | # (RP*VP)*RP              | 48 |
| 49 |         | DOT   | DMP    |                           | 49 |
| 50 |         |       | VPASS3 |                           | 50 |
| 51 |         |       | 14D    |                           | 51 |
| 52 |         |       |        |                           | 52 |
| 53 |         |       |        |                           | 53 |
| 54 |         |       |        |                           | 54 |
| 55 |         |       |        |                           | 55 |
| 56 |         |       |        |                           | 56 |
| 57 |         |       |        |                           | 57 |
| 58 |         |       |        |                           | 58 |
| 59 |         |       |        |                           | 59 |
| 60 |         |       |        |                           | 60 |

|    |                 |         |                                     |    |
|----|-----------------|---------|-------------------------------------|----|
| 1  | # 101 051 11 12 |         |                                     | 1  |
| 2  |                 |         |                                     | 2  |
| 3  | BDSU            |         |                                     | 3  |
| 4  | NORM            | PDVL    | # NORMALIZED WA - WP 12D            | 4  |
| 5  |                 | X1      |                                     | 5  |
| 6  |                 | 6D      |                                     | 6  |
| 7  | VXV             | DOT     |                                     | 7  |
| 8  |                 | OD      |                                     | 8  |
| 9  | PDVL            | UNRM    | # RA*RP.UN 14D                      | 9  |
| 10 |                 | DOT     |                                     | 10 |
| 11 |                 | OD      |                                     | 11 |
| 12 | SL1             | 6D      |                                     | 12 |
| 13 | SIGN            | ACOS    |                                     | 13 |
| 14 | DSU             | DAD     | # ALPHA PI                          | 14 |
| 15 |                 | DPHALF  |                                     | 15 |
| 16 |                 | ELEV    |                                     | 16 |
| 17 | PDDL            | ACOS    |                                     | 17 |
| 18 |                 | 28D     |                                     | 18 |
| 19 | BDSU            | SIGN    |                                     | 19 |
| 20 |                 | DPHALF  |                                     | 20 |
| 21 |                 | 30D     | # CONTAINS RP-RA                    | 21 |
| 22 | DAD             |         |                                     | 22 |
| 23 | DMP             | DDV     |                                     | 23 |
| 24 |                 | TWOPI   |                                     | 24 |
| 25 | DMP             |         |                                     | 25 |
| 26 | SL*             | DMP     |                                     | 26 |
| 27 |                 | 0       | -3,1                                | 27 |
| 28 | PUSH            | ABS     |                                     | 28 |
| 29 | DSU             | BMN     |                                     | 29 |
| 30 |                 | SECMAX  |                                     | 30 |
| 31 |                 | OKMAX   |                                     | 31 |
| 32 | DLOAD           | SIGN    | # REPLACE TIME WITH MAX TIME SIGNED | 32 |
| 33 |                 | SECMAX  |                                     | 33 |
| 34 | OKMAX           | PUSH    |                                     | 34 |
| 35 |                 | SLOAD   | # TEST FIRST ITERATION              | 35 |
| 36 |                 | BPL     |                                     | 36 |
| 37 |                 | TITER   |                                     | 37 |
| 38 | SSP             | REPETE  |                                     | 38 |
| 39 |                 | DLOAD   |                                     | 39 |
| 40 |                 | TITER   |                                     | 40 |
| 41 | OCT             | 37777   |                                     | 41 |
| 42 | GOTO            |         |                                     | 42 |
| 43 | REPETE          | STORDEL |                                     | 43 |
| 44 |                 | DMP     |                                     | 44 |
| 45 |                 | DELEL   |                                     | 45 |
| 46 |                 | DELELO  |                                     | 46 |
| 47 | BPL             | DLOAD   |                                     | 47 |
| 48 |                 | NEXTES  |                                     | 48 |
| 49 |                 | SECMAX  |                                     | 49 |
| 50 | DMP             |         |                                     | 50 |
| 51 | STODL           | THIRD   |                                     | 51 |
| 52 |                 | SECMAX  |                                     | 52 |
| 53 |                 |         |                                     | 53 |
| 54 |                 |         |                                     | 54 |
| 55 |                 |         |                                     | 55 |
| 56 |                 |         |                                     | 56 |
| 57 |                 |         |                                     | 57 |
| 58 |                 |         |                                     | 58 |
| 59 |                 |         |                                     | 59 |
| 60 |                 |         |                                     | 60 |

|    |                |       |          |                           |    |
|----|----------------|-------|----------|---------------------------|----|
| 1  | # 151 55111 15 |       |          | PAGE 010                  | 1  |
| 2  |                | ABS   | SR1      | # CROSSED OVER SOLUTION   | 2  |
| 3  |                | DCOMP | GOTO     | # DT=(-SIGN(DTO)//DT//)/2 | 3  |
| 4  |                |       | RESIGN   |                           | 4  |
| 5  | NEXTES         | DLOAD | ABS      |                           | 5  |
| 6  |                |       | DELEL    |                           | 6  |
| 7  |                | PDDL  | ABS      |                           | 7  |
| 8  |                |       | DELELO   |                           | 8  |
| 9  |                | DSU   |          |                           | 9  |
| 10 |                | BMN   | DLOAD    |                           | 10 |
| 11 |                |       | REVERS   | # WRONG DIRECTION         | 11 |
| 12 |                | ABS   |          |                           | 12 |
| 13 | RESIGN         | SIGN  | GOTO     |                           | 13 |
| 14 |                |       | DELTEEO  |                           | 14 |
| 15 |                |       | STORDELT |                           | 15 |
| 16 | REVERS         | DLOAD | DCOMP    |                           | 16 |
| 17 |                |       | DELTEEO  |                           | 17 |
| 18 |                | PUSH  | SR1      |                           | 18 |
| 19 |                | STORE | DELTEEO  |                           | 19 |
| 20 |                | DAD   |          |                           | 20 |
| 21 |                | GOTO  |          |                           | 21 |
| 22 |                |       | ADTIME   |                           | 22 |
| 23 | STORDELT       | STORE | DELTEEO  |                           | 23 |
| 24 | ADTIME         | DAD   |          |                           | 24 |
| 25 |                |       | NOMTPI   | # SUM OF DELTA T'S        | 25 |
| 26 |                | STORE | NOMTPI   |                           | 26 |
| 27 |                | VLOAD | PDVL     |                           | 27 |
| 28 |                |       | VAPREC   |                           | 28 |
| 29 |                |       | RAPREC   |                           | 29 |
| 30 |                | CALL  |          |                           | 30 |
| 31 |                |       | GOINT    |                           | 31 |
| 32 |                | CALL  |          |                           | 32 |
| 33 |                |       | ACTIVE   | # STORE NEW RACT3 VACT3   | 33 |
| 34 |                | VLOAD | PDVL     |                           | 34 |
| 35 |                |       | VPPREC   |                           | 35 |
| 36 |                |       | RPPREC   |                           | 36 |
| 37 |                | CALL  |          |                           | 37 |
| 38 |                |       | GOINT    |                           | 38 |
| 39 |                | CALL  |          |                           | 39 |
| 40 |                |       | PASSIVE  | # STORE NEW RPASS3 VPASS3 | 40 |
| 41 |                | GOTO  |          |                           | 41 |
| 42 |                |       | ELCALC   |                           | 42 |
| 43 | ELEX           | DLOAD | DAD      |                           | 43 |
| 44 |                |       | TTPI     |                           | 44 |
| 45 |                |       | NOMTPI   |                           | 45 |
| 46 |                | STODL | TTPI     |                           | 46 |
| 47 |                | BON   |          |                           | 47 |
| 48 |                |       | ETPIFLAG |                           | 48 |
| 49 |                |       | TIMEX    |                           | 49 |
| 50 |                | STORE | ELEV     |                           | 50 |
| 51 | TIMEX          | DLOAD | GOTO     |                           | 51 |
| 52 |                |       |          |                           | 52 |
| 53 |                |       |          |                           | 53 |
| 54 |                |       |          |                           | 54 |
| 55 |                |       |          |                           | 55 |
| 56 |                |       |          |                           | 56 |
| 57 |                |       |          |                           | 57 |
| 58 |                |       |          |                           | 58 |
| 59 |                |       |          |                           | 59 |
| 60 |                |       |          |                           | 60 |



|    |                    |    |
|----|--------------------|----|
| 1  | ZEROVECS<br>NORMEX | 1  |
| 2  |                    | 2  |
| 3  |                    | 3  |
| 4  |                    | 4  |
| 5  |                    | 5  |
| 6  |                    | 6  |
| 7  |                    | 7  |
| 8  |                    | 8  |
| 9  |                    | 9  |
| 10 |                    | 10 |
| 11 |                    | 11 |
| 12 |                    | 12 |
| 13 |                    | 13 |
| 14 |                    | 14 |
| 15 |                    | 15 |
| 16 |                    | 16 |
| 17 |                    | 17 |
| 18 |                    | 18 |
| 19 |                    | 19 |
| 20 |                    | 20 |
| 21 |                    | 21 |
| 22 |                    | 22 |
| 23 |                    | 23 |
| 24 |                    | 24 |
| 25 |                    | 25 |
| 26 |                    | 26 |
| 27 |                    | 27 |
| 28 |                    | 28 |
| 29 |                    | 29 |
| 30 |                    | 30 |
| 31 |                    | 31 |
| 32 |                    | 32 |
| 33 |                    | 33 |
| 34 |                    | 34 |
| 35 |                    | 35 |
| 36 |                    | 36 |
| 37 |                    | 37 |
| 38 |                    | 38 |
| 39 |                    | 39 |
| 40 |                    | 40 |
| 41 |                    | 41 |
| 42 |                    | 42 |
| 43 |                    | 43 |
| 44 |                    | 44 |
| 45 |                    | 45 |
| 46 |                    | 46 |
| 47 |                    | 47 |
| 48 |                    | 48 |
| 49 |                    | 49 |
| 50 |                    | 50 |
| 51 |                    | 51 |
| 52 |                    | 52 |
| 53 |                    | 53 |
| 54 |                    | 54 |
| 55 |                    | 55 |
| 56 |                    | 56 |
| 57 |                    | 57 |
| 58 |                    | 58 |
| 59 |                    | 59 |
| 60 |                    | 60 |



|    |                                                                      |       |        |    |
|----|----------------------------------------------------------------------|-------|--------|----|
| 1  | # ***** S34/35.1 *****                                               |       |        | 1  |
| 2  |                                                                      |       |        | 2  |
| 3  |                                                                      |       |        | 3  |
| 4  | # COMPUTE UNIT NORMAL AND LINE OF SIGHT VECTORS GIVEN THE ACTIVE AND |       |        | 4  |
| 5  | # PASSIVE POS AND VEL AT TIME T3                                     |       |        | 5  |
| 6  |                                                                      |       |        | 6  |
| 7  | S34/35.1                                                             | VLOAD | VSU    | 7  |
| 8  |                                                                      |       | RPASS3 | 8  |
| 9  |                                                                      |       | RACT3  | 9  |
| 10 |                                                                      | UNIT  | PUSH   | 10 |
| 11 |                                                                      | STOVL | ULOS   | 11 |
| 12 |                                                                      |       | RACT3  | 12 |
| 13 |                                                                      | VXV   | UNIT   | 13 |
| 14 |                                                                      |       | VACT3  | 14 |
| 15 |                                                                      | STORE | UNRM   | 15 |
| 16 |                                                                      | RVQ   |        | 16 |
| 17 |                                                                      |       |        | 17 |
| 18 |                                                                      |       |        | 18 |
| 19 |                                                                      |       |        | 19 |
| 20 |                                                                      |       |        | 20 |
| 21 |                                                                      |       |        | 21 |
| 22 |                                                                      |       |        | 22 |
| 23 |                                                                      |       |        | 23 |
| 24 |                                                                      |       |        | 24 |
| 25 |                                                                      |       |        | 25 |
| 26 |                                                                      |       |        | 26 |
| 27 |                                                                      |       |        | 27 |
| 28 |                                                                      |       |        | 28 |
| 29 |                                                                      |       |        | 29 |
| 30 |                                                                      |       |        | 30 |
| 31 |                                                                      |       |        | 31 |
| 32 |                                                                      |       |        | 32 |
| 33 |                                                                      |       |        | 33 |
| 34 |                                                                      |       |        | 34 |
| 35 |                                                                      |       |        | 35 |
| 36 |                                                                      |       |        | 36 |
| 37 |                                                                      |       |        | 37 |
| 38 |                                                                      |       |        | 38 |
| 39 |                                                                      |       |        | 39 |
| 40 |                                                                      |       |        | 40 |
| 41 |                                                                      |       |        | 41 |
| 42 |                                                                      |       |        | 42 |
| 43 |                                                                      |       |        | 43 |
| 44 |                                                                      |       |        | 44 |
| 45 |                                                                      |       |        | 45 |
| 46 |                                                                      |       |        | 46 |
| 47 |                                                                      |       |        | 47 |
| 48 |                                                                      |       |        | 48 |
| 49 |                                                                      |       |        | 49 |
| 50 |                                                                      |       |        | 50 |
| 51 |                                                                      |       |        | 51 |
| 52 |                                                                      |       |        | 52 |
| 53 |                                                                      |       |        | 53 |
| 54 |                                                                      |       |        | 54 |
| 55 |                                                                      |       |        | 55 |
| 56 |                                                                      |       |        | 56 |
| 57 |                                                                      |       |        | 57 |
| 58 |                                                                      |       |        | 58 |
| 59 |                                                                      |       |        | 59 |
| 60 |                                                                      |       |        | 60 |

# \*\*\*\*\* S34/35.2 \*\*\*\*\*

# ADVANCE PASSIVE VEH TO RENDEZVOUS TIME AND GET REQ VEL FROM LAMBERT

S34/35.2 STQ VLOAD

SUBEXIT

VPASS3

PDVL PDDL

RPASS3

INTIME

PDDL PDDL

TPASS4

TWOPI

# CONIC

PDDL BHIZ

NN

S3435.23

DLOAD

DLOAD

PUSH

ZEROVECS

# PRECISION

S3435.23

CALL

INTINT

# GET TARGET VECTOR

S3435.25

STOVL

RTARG

VATT

STOVL

VPASS4

RTARG

# COMPUTE PHI = PI + (ACOS(UNIT RA.UNIT RP) - PI)SIGN(RA\*RP.U)

UNIT

PDVL

# UNIT RP

RACT3

UNIT

PUSH

# UNIT RA

VXV

DOT

OD

UNRM

# RA\*RP.U

PDVL

DOT

SL1

# UNIT RA.UNIT RP

OD

ACOS

SIGN

BPL

DAD

NOPIE

NOPIE

STODL

DPPOSMAX

# REASONABLE TWO PI

ACTCENT

TPASS4

DSU

INTIME

STORE

DELLT4

SLOAD

SETPD

NN

# NUMBER OF OFFSETS

OD

PDDL

PDVL

EPSFOUR

RACT3

STOVL

RINIT

|                |                                |
|----------------|--------------------------------|
| STCALL         | VACT3<br>VINIT                 |
| CALL           | INITVEL                        |
| VLOAD          | LOMAT<br>MXV<br>DELVEET3<br>OD |
| VSL1<br>STCALL | DELVLVC<br>SUBEXIT             |



# \*\*\*\*\* S34/35.3 \*\*\*\*\*

|          |                      |                         |                             |
|----------|----------------------|-------------------------|-----------------------------|
| S34/35.3 | STQ                  | CALL<br>NORMEX<br>LOMAT | # GET MATRIX IN PUSH LIST   |
|          | VLOAD                | VXM<br>DELVLVC<br>OD    | # NEW DEL V TPI             |
|          | VSL1<br>STORE<br>VAD | DELVEET3<br>PDVL        | # SAVE FOR TRANSFORM        |
|          |                      | VACT3<br>RACT3          | # NEW V REQ                 |
|          | PDDL                 | PDDL                    |                             |
|          |                      | TIG<br>TPASS4<br>PUSH   |                             |
|          | PDDL                 | DPPOSMAX                |                             |
|          | CALL                 | INTINT                  | # INTEG. FOR NEW TARGET VEC |
|          | VLOAD                |                         |                             |
|          |                      | RATT<br>RTARG           |                             |
| NOVRWRT  | STORE<br>VLOAD       | PUSH<br>ULOS            |                             |
|          | VXV                  | VCOMP                   |                             |
|          |                      | UNRM                    |                             |
|          | UNIT<br>VXV          | PUSH<br>VSL1            |                             |
|          |                      | ULOS                    |                             |
|          | PDVL<br>PDVL         | MXV                     |                             |
|          |                      | DELVEET3<br>OD          |                             |
|          | VSL1<br>STCALL       | DVLOS<br>NORMEX         |                             |



|    |                        |      |         |                          |    |
|----|------------------------|------|---------|--------------------------|----|
| 1  | # ***** S34/35.4 ***** |      |         |                          | 1  |
| 2  |                        |      |         |                          | 2  |
| 3  |                        |      |         |                          | 3  |
| 4  | S34/35.4               | STQ  | SETPD   | # NO ASTRONAUT OVERWRITE | 4  |
| 5  |                        |      | NORMEX  |                          | 5  |
| 6  |                        |      | OD      |                          | 6  |
| 7  |                        | GOTO |         |                          | 7  |
| 8  |                        |      | NOVRWRT |                          | 8  |
| 9  |                        |      |         |                          | 9  |
| 10 |                        |      |         |                          | 10 |
| 11 |                        |      |         |                          | 11 |
| 12 |                        |      |         |                          | 12 |
| 13 |                        |      |         |                          | 13 |
| 14 |                        |      |         |                          | 14 |
| 15 |                        |      |         |                          | 15 |
| 16 |                        |      |         |                          | 16 |
| 17 |                        |      |         |                          | 17 |
| 18 |                        |      |         |                          | 18 |
| 19 |                        |      |         |                          | 19 |
| 20 |                        |      |         |                          | 20 |
| 21 |                        |      |         |                          | 21 |
| 22 |                        |      |         |                          | 22 |
| 23 |                        |      |         |                          | 23 |
| 24 |                        |      |         |                          | 24 |
| 25 |                        |      |         |                          | 25 |
| 26 |                        |      |         |                          | 26 |
| 27 |                        |      |         |                          | 27 |
| 28 |                        |      |         |                          | 28 |
| 29 |                        |      |         |                          | 29 |
| 30 |                        |      |         |                          | 30 |
| 31 |                        |      |         |                          | 31 |
| 32 |                        |      |         |                          | 32 |
| 33 |                        |      |         |                          | 33 |
| 34 |                        |      |         |                          | 34 |
| 35 |                        |      |         |                          | 35 |
| 36 |                        |      |         |                          | 36 |
| 37 |                        |      |         |                          | 37 |
| 38 |                        |      |         |                          | 38 |
| 39 |                        |      |         |                          | 39 |
| 40 |                        |      |         |                          | 40 |
| 41 |                        |      |         |                          | 41 |
| 42 |                        |      |         |                          | 42 |
| 43 |                        |      |         |                          | 43 |
| 44 |                        |      |         |                          | 44 |
| 45 |                        |      |         |                          | 45 |
| 46 |                        |      |         |                          | 46 |
| 47 |                        |      |         |                          | 47 |
| 48 |                        |      |         |                          | 48 |
| 49 |                        |      |         |                          | 49 |
| 50 |                        |      |         |                          | 50 |
| 51 |                        |      |         |                          | 51 |
| 52 |                        |      |         |                          | 52 |
| 53 |                        |      |         |                          | 53 |
| 54 |                        |      |         |                          | 54 |
| 55 |                        |      |         |                          | 55 |
| 56 |                        |      |         |                          | 56 |
| 57 |                        |      |         |                          | 57 |
| 58 |                        |      |         |                          | 58 |
| 59 |                        |      |         |                          | 59 |
| 60 |                        |      |         |                          | 60 |

# \*\*\*\*\* LOMAT \*\*\*\*\*

LOMAT

VLOAD

VCOMP

STOVL

6D

# Y

UNIT

STORE

RACT3

VCOMP

12D

VXV

VSL1

STORE

0D

# Z\*-Y

SETPD

RVQ

PDDL

PDDL

# DO

GOINT

ZEROVECS

#

NOT

PUSH

PUSH

#

ORDER OR INSERT BEFORE INTINT

INTINT

STQ

CALL

RTRN

INTSTALL

CLEAR

DLOAD

BZE

INTYPFLG

SET

+2

DLOAD

STADR

STODL

TDEC1

SET

LXA,2

MOONFLAG

BON

RTX2

CLEAR

CMOONFLG

ALLSET

STOVL

ALLSET

VSR\*

MOONFLAG

VSR\*

TET

STCALL

0,2

VCV

VLOAD

INTEGRVS

GOTO

RATT

RTRN

# \*\*\*\*\* S34/35.5 \*\*\*\*\*  
# SUBROUTINES USED

# BANKCALL  
# GOFLASH  
# GOTOP00H  
# S34/35.3  
# S34.35.4  
# VNPOOH

S34/35.5 STQ BON  
SUBEXIT

SET  
FINALFLG  
FLAGON  
GOTO

FLAGON CLEAR  
UPDATFLG  
FLAGOFF  
VLOAD

STORE  
NTARGFLG  
DEVLVLC  
GDT/2

+5 EXIT  
CAF V06N81  
TC BANKCALL

CADR GOFLASH  
TC GOTOP00H  
TC +2 # PRO

+2 TC FLAGON +5 # LOAD  
CA EBANK7  
TS EBANK # TO BE SURE

NTARGCHK ZL  
CA FIVE

TS Q  
INDEX Q  
CS DELVLVC

INDEX Q  
AD GDT/2  
ADS L

CCS Q  
TCF NTARGCHK  
LXCH A

EXTEND  
BZF +3  
TC UPFLAG

ADRES NTARGFLG

TC INTPRET  
BOFF CALL  
NTARGFLG

|    |         |       |          |    |
|----|---------|-------|----------|----|
| 1  |         |       |          | 1  |
| 2  |         |       |          | 2  |
| 3  |         |       |          | 3  |
| 4  | NOCHG   | CLEAR | NOCHG    | 4  |
| 5  |         |       | S34/35.3 | 5  |
| 6  |         |       | VLOAD    | 6  |
| 7  |         |       | XDELVFLG | 7  |
| 8  |         |       | DELVEET3 | 8  |
| 9  | FLAGOFF | STORE | DELVSIN  | 9  |
| 10 |         |       | S34/35.4 | 10 |
| 11 |         |       | EXIT     | 11 |
| 12 |         |       | CAF      | 12 |
| 13 |         |       | TC       | 13 |
| 14 |         |       | V06N59   | 14 |
| 15 |         |       | VNPOOH   | 15 |
| 16 |         |       | INTPRET  | 16 |
| 17 |         |       | SUBEXIT  | 17 |
| 18 |         |       |          | 18 |
| 19 |         |       |          | 19 |
| 20 |         |       |          | 20 |
| 21 |         |       |          | 21 |
| 22 |         |       |          | 22 |
| 23 |         |       |          | 23 |
| 24 |         |       |          | 24 |
| 25 |         |       |          | 25 |
| 26 |         |       |          | 26 |
| 27 |         |       |          | 27 |
| 28 |         |       |          | 28 |
| 29 |         |       |          | 29 |
| 30 |         |       |          | 30 |
| 31 |         |       |          | 31 |
| 32 |         |       |          | 32 |
| 33 |         |       |          | 33 |
| 34 |         |       |          | 34 |
| 35 |         |       |          | 35 |
| 36 |         |       |          | 36 |
| 37 |         |       |          | 37 |
| 38 |         |       |          | 38 |
| 39 |         |       |          | 39 |
| 40 |         |       |          | 40 |
| 41 |         |       |          | 41 |
| 42 |         |       |          | 42 |
| 43 |         |       |          | 43 |
| 44 |         |       |          | 44 |
| 45 |         |       |          | 45 |
| 46 |         |       |          | 46 |
| 47 |         |       |          | 47 |
| 48 |         |       |          | 48 |
| 49 |         |       |          | 49 |
| 50 |         |       |          | 50 |
| 51 |         |       |          | 51 |
| 52 |         |       |          | 52 |
| 53 |         |       |          | 53 |
| 54 |         |       |          | 54 |
| 55 |         |       |          | 55 |
| 56 |         |       |          | 56 |
| 57 |         |       |          | 57 |
| 58 |         |       |          | 58 |
| 59 |         |       |          | 59 |
| 60 |         |       |          | 60 |

# \*\*\*\*\* VN1645 \*\*\*\*\*

#

# SUBROUTINES USED

# P3XORP7X

# GET+MGA

# BANKCALL

# DELAYJOB

# COMPTGO

# GOFLASHR

# GOTOP00H

# FLAGUP

VN1645

STQ

DLOAD

SUBEXIT

DP-.01

STORE

+MGA

# MGA = -.01

BOFF

DLOAD

FINALFLG

GET45

DP-.01

DAD

DP-.01

STORE

+MGA

# MGA = -.02

BOFF

EXIT

REFSMFLG

GET45

TC

P3XORP7X

TC

+2

# P3X

TC

GET45 +1

# P7X

TC

INTPRET

VLOAD

PUSH

DELVSIN

CALL

# COMPUTE MGA

GET+MGA

GET45

EXIT

COMPTGO

# INITIATE TASK TO UPDATE TTOGO

CA

SUBEXIT

TS

QSAVED

CAF

1SEC

TC

BANKCALL

CADR

DELAYJOB

CAF

V16N45

# TRKMKCNT, TTOGO, +MGA

TC

BANKCALL

CADR

GOFLASH

TC

KILCLOCK

# TERMINATE

TC

N45PROC

# PROCEED

TC

CLUPDATE

# RECYCLE -- RETURN FOR INITIAL COMPUTATION

KILCLOCK

CA

Z

TS

DISPDEX

[illegible]



|    |                        |          |          |    |
|----|------------------------|----------|----------|----|
| 1  |                        |          |          | 1  |
| 2  | # ***** DISPLAYE ***** |          |          | 2  |
| 3  | #                      |          |          | 3  |
| 4  | # SUBROUTINES USED     |          |          | 4  |
| 5  | #                      | BANKCALL |          | 5  |
| 6  | #                      | GOFLASHR |          | 6  |
| 7  | #                      | GOTOP00H |          | 7  |
| 8  | #                      | BLANKET  |          | 8  |
| 9  | #                      | ENDOFJOB |          | 9  |
| 10 |                        |          |          | 10 |
| 11 | DISPLAYE               | EXTEND   |          | 11 |
| 12 |                        | QXCH     | NORMEX   | 12 |
| 13 |                        | CAF      | V06N55   | 13 |
| 14 |                        | TCR      | BANKCALL | 14 |
| 15 |                        | CADR     | GOFLASH  | 15 |
| 16 |                        | TCF      | GOTOP00H | 16 |
| 17 |                        | TC       | NORMEX   | 17 |
| 18 |                        | TCF      | -5       | 18 |
| 19 |                        |          |          | 19 |
| 20 |                        |          |          | 20 |
| 21 |                        |          |          | 21 |
| 22 |                        |          |          | 22 |
| 23 |                        |          |          | 23 |
| 24 |                        |          |          | 24 |
| 25 |                        |          |          | 25 |
| 26 |                        |          |          | 26 |
| 27 |                        |          |          | 27 |
| 28 |                        |          |          | 28 |
| 29 |                        |          |          | 29 |
| 30 |                        |          |          | 30 |
| 31 |                        |          |          | 31 |
| 32 |                        |          |          | 32 |
| 33 |                        |          |          | 33 |
| 34 |                        |          |          | 34 |
| 35 |                        |          |          | 35 |
| 36 |                        |          |          | 36 |
| 37 |                        |          |          | 37 |
| 38 |                        |          |          | 38 |
| 39 |                        |          |          | 39 |
| 40 |                        |          |          | 40 |
| 41 |                        |          |          | 41 |
| 42 |                        |          |          | 42 |
| 43 |                        |          |          | 43 |
| 44 |                        |          |          | 44 |
| 45 |                        |          |          | 45 |
| 46 |                        |          |          | 46 |
| 47 |                        |          |          | 47 |
| 48 |                        |          |          | 48 |
| 49 |                        |          |          | 49 |
| 50 |                        |          |          | 50 |
| 51 |                        |          |          | 51 |
| 52 |                        |          |          | 52 |
| 53 |                        |          |          | 53 |
| 54 |                        |          |          | 54 |
| 55 |                        |          |          | 55 |
| 56 |                        |          |          | 56 |
| 57 |                        |          |          | 57 |
| 58 |                        |          |          | 58 |
| 59 |                        |          |          | 59 |
| 60 |                        |          |          | 60 |





# \*\*\*\*\* P3XORP7X \*\*\*\*\*

|          |        |        |
|----------|--------|--------|
| P3XORP7X | CAF    | HIGH9  |
|          | MASK   | MODREG |
|          | EXTEND |        |
|          | BZF    | +2     |
|          | INCR   | Q      |
|          | RETURN |        |

# \*\*\*\*\* VNPOOH \*\*\*\*\*

#

# SUBROUTINES USED

# BANKCALL

# GOFLASH

# GOTOPOOH

|        |        |          |
|--------|--------|----------|
| VNPOOH | EXTEND |          |
|        | QXCH   | RTRN     |
|        | TS     | VERBNOUN |
|        | CA     | VERBNOUN |
|        | TCR    | BANKCALL |
|        | CADR   | GOFLASH  |
|        | TCF    | GOTOPOOH |
|        | TC     | RTRN     |
|        | TCF    | -5       |

# \*\*\*\*\* CONSTANTS \*\*\*\*\*

|         |      |             |                  |
|---------|------|-------------|------------------|
| V06N37  | VN   | 0637        |                  |
| V06N55  | VN   | 0655        |                  |
| V06N58  | VN   | 0658        |                  |
| V06N59  | VN   | 0659        |                  |
| V06N81  | VN   | 0681        |                  |
| V16N45  | VN   | 1645        |                  |
| TWOPI   | 2DEC | 6.283185307 | B-4              |
| MAX250  | 2DEC | 25          | E3               |
| THIRD   | 2DEC | .333333333  |                  |
| ELEPS   | 2DEC | .27777777   | E-3              |
| DP-.01  | OCT  | 77777       | # CONSTANTS      |
|         | OCT  | 61337       | # ADJACENT       |
| EPSFOUR | 2DEC | .0416666666 | -.01 FOR MGA DSP |
| 130DEG  | 2DEC | .3611111111 |                  |

# \*\*\*\*\* INITVEL \*\*\*\*\*

# MOD NO -1 LOG SECTION -- P34-P35, P74-P75  
# MOD BY WHITE, P. DATE: 21 NOV 67

# FUNCTIONAL

# THIS SUBROUTINE COMPUTES THE REQUIRED INITIAL VELOCITY VECTOR FOR  
# A TRAJECTORY OF SPECIFIC TRANSFER TIME BETWEEN SPECIFIED INITIAL  
# AND TARGET POSITIONS. THE TRAJECTORY MAY BE EITHER CONIC OR  
# PRECISION DEPENDING ON AN INPUT PARAMETER (NAMELY, NUMBER OF  
# OFFSETS). IN ADDITION, IN THE PRECISION TRAJECTORY CASE, THE  
# SUBROUTINE ALSO COMPUTES AN OFFSET TARGET VECTOR, TO BE USED  
# DURING PURE-CONIC CROSS-PRODUCT STEERING. THE OFFSET TARGET  
# VECTOR IS THE TERMINAL POSITION VECTOR OF A CONIC TRAJECTORY WHICH  
# HAS THE SAME INITIAL STATE AS A PRECISION TRAJECTORY WHOSE  
# TERMINAL POSITION VECTOR IS THE SPECIFIED TARGET VECTOR.# IN ORDER TO AVOID THE INHERENT SINGULARITIES IN THE 180 DEGREE  
# TRANSFER CASE WHEN THE (TRUE OR OFFSET) TARGET VECTOR MAY BE  
# SLIGHTLY OUT OF THE ORBITAL PLANE, THIS SUBROUTINE ROTATES THIS  
# VECTOR INTO A PLANE DEFINED BY THE INPUT INITIAL POSITION VECTOR  
# AND ANOTHER INPUT VECTOR (USUALLY THE INITIAL VELOCITY VECTOR),  
# WHENEVER THE INPUT TARGET VECTOR LIES INSIDE A CONE WHOSE VERTEX  
# IS THE ORIGIN OF COORDINATES, WHOSE AXIS IS THE 180 DEGREE  
# TRANSFER DIRECTION, AND WHOSE CONE ANGLE IS SPECIFIED BY THE USER.# THE LAMBERT SUBROUTINE IS UTILIZED FOR THE CONIC COMPUTATIONS AND  
# THE COASTING INTEGRATION SUBROUTINE IS UTILIZED FOR THE PRECISION  
# TRAJECTORY COMPUTATIONS.

# CALLING SEQUENCE

# L CALL  
# L+1 INITVEL  
# L+2 (RETURN -- ALWAYS)

# INPUT

# (1) RINIT INITIAL POSITION RADIUS VECTOR  
# (2) VINIT INITIAL POSITION VELOCITY VECTOR  
# (3) RTARG TARGET POSITION RADIUS VECTOR  
# (4) DELLT4 DESIRED TIME OF FLIGHT FROM RINIT TO RTARG  
# (5) INTIME TIME OF RINIT  
# (6) OD NUMBER OF ITERATIONS OF LAMBERT/INTEGRVS  
# (7) 2D ANGLE TO 180 DEGREES WHEN ROTATION STARTS  
# (8) RTX1 -2 FOR EARTH, -10D FOR LUNAR  
# (9) RTX2 COORDINATE SYSTEM ORIGIN -- 0 FOR EARTH, 2 FOR LUNAR  
# PUSHLOC SET AT 4D

# OUTPUT

|   |     |          |                                      |
|---|-----|----------|--------------------------------------|
| # | (1) | RTARG    | OFFSET TARGET POSITION VECTOR        |
| # | (2) | VIPRIME  | MANEUVER VELOCITY REQUIRED           |
| # | (3) | VTPRIME  | VELOCITY AT TARGET AFTER MANEUVER    |
| # | (4) | DELVEET3 | DELTA VELOCITY REQUIRED FOR MANEUVER |

# SUBROUTINES USED

|   |          |
|---|----------|
| # | LAMBERT  |
| # | INTSTALL |
| # | INTEGRVS |

|        |        |
|--------|--------|
| SETLOC | INTVEL |
| BANK   |        |

|        |            |
|--------|------------|
| COUNT* | \$\$/INITV |
|--------|------------|

|         |     |                            |
|---------|-----|----------------------------|
| INITVEL | SET | # COGA GUESS NOT AVAILABLE |
|---------|-----|----------------------------|

|          |       |        |
|----------|-------|--------|
| HAVEGUES | VLOAD | GUESSW |
|          |       | STQ    |

|       |        |
|-------|--------|
|       | RTARG  |
|       | NORMEX |
| STORE | RTARG1 |

|       |       |
|-------|-------|
| ABVAL |       |
| STORE | RTMAG |
| SLOAD | BHIZ  |

|       |          |
|-------|----------|
|       | RTX2     |
|       | INITVEL1 |
| VLOAD | VSL2     |

|       |       |       |
|-------|-------|-------|
|       | RINIT | # B29 |
| STOVL | RINIT | # B27 |
|       | VINIT | # B7  |

|       |        |
|-------|--------|
| VSL2  |        |
| STOVL | VINIT  |
|       | RTARG1 |

|       |        |
|-------|--------|
| VSL2  |        |
| STORE | RTARG1 |
| ABVAL |        |
| STORE | RTMAG  |

# INITIALIZATION

|          |     |       |                                             |
|----------|-----|-------|---------------------------------------------|
| INITVEL1 | SSP | DLOAD | # SET ITCTR TO -1,LOAD MPAC WITH E4 (PL 2D) |
|----------|-----|-------|---------------------------------------------|

|  |       |   |    |
|--|-------|---|----|
|  | ITCTR | 0 | -1 |
|--|-------|---|----|

|        |       |                                    |
|--------|-------|------------------------------------|
| COSINE | SR1   | # CALCULATE COSINE (E4) (+2)       |
| STODL  | COZY4 | # SET COZY4 TO COSINE (E4) (PL 0D) |

|       |           |                            |
|-------|-----------|----------------------------|
| LXA,2 | SXA,2     |                            |
|       | MPAC      |                            |
|       | VTARGETAG | # SET VTARGETAG TO 0D (SP) |

|       |  |
|-------|--|
| VLOAD |  |
|-------|--|

|    |                                                    |          |                                     |          |
|----|----------------------------------------------------|----------|-------------------------------------|----------|
| 1  |                                                    | RINIT    |                                     |          |
| 2  |                                                    | R1VEC    | # R1VEC EQ RINIT                    |          |
| 3  | STOVL                                              | RTARG1   |                                     |          |
| 4  |                                                    | R2VEC    | # R2VEC EQ RTARG                    |          |
| 5  | STODL                                              | DELLT4   |                                     |          |
| 6  |                                                    | TDESIRED | # TDESIRED EQ DELLT4                |          |
| 7  | STORE                                              | VLOAD    |                                     |          |
| 8  | SETPD                                              | OD       | # INITIALIZE PL TO OD               |          |
| 9  |                                                    | RINIT    | # MPAC EQ RINIT (+29)               |          |
| 10 | UNIT                                               | PUSH     | # UNIT(RI) (+1)                     | (PL 6D)  |
| 11 | VXV                                                | UNIT     |                                     |          |
| 12 |                                                    | VINIT    | # MPAC EQ UNIT(RI) X VI (+8)        |          |
| 13 | STOVL                                              | UN       |                                     |          |
| 14 |                                                    | RTARG1   |                                     |          |
| 15 |                                                    | DOT      | # TEMP*RT.URI (+2)                  | (PL 0D)  |
| 16 | UNIT                                               | CLEAR    |                                     |          |
| 17 | DAD                                                | COZY4    |                                     |          |
| 18 |                                                    | NORMSW   |                                     |          |
| 19 |                                                    | COZY4    |                                     |          |
| 20 | STORE                                              | SET      |                                     |          |
| 21 | INITVEL2                                           | BPL      |                                     |          |
| 22 |                                                    | INITVEL3 | # UN CALCULATED IN LAMBERT          |          |
| 23 |                                                    | NORMSW   |                                     |          |
| 24 |                                                    |          |                                     |          |
| 25 | # ROTATE RC INTO YC PLANE -- SET UNIT NORMAL TO YC |          |                                     |          |
| 26 |                                                    |          |                                     |          |
| 27 | VLOAD                                              | PUSH     | #                                   | (PL 6D)  |
| 28 |                                                    | R2VEC    | # RC TO 6D (+29)                    |          |
| 29 | ABVAL                                              | PDVL     | # RC TO MPAC, ABVAL(RC) (+29) TO OD | (PL 2D)  |
| 30 | PUSH                                               | VPROJ    | #                                   | (PL 8D)  |
| 31 |                                                    | UN       |                                     |          |
| 32 | VSL2                                               | BVSU     |                                     |          |
| 33 | UNIT                                               | VXSC     | #                                   | (PL 0D)  |
| 34 | VSL1                                               |          |                                     |          |
| 35 | STORE                                              | R2VEC    |                                     |          |
| 36 | TLOAD                                              | SLOAD    |                                     |          |
| 37 |                                                    | ZEROVEC  |                                     |          |
| 38 |                                                    | ITCTR    |                                     |          |
| 39 | BPL                                                | VLOAD    |                                     |          |
| 40 |                                                    | INITVEL3 |                                     |          |
| 41 |                                                    | R2VEC    |                                     |          |
| 42 | STORE                                              | RTARG1   |                                     |          |
| 43 | INITVEL3                                           | DLOAD    | #                                   | (PL 2D)  |
| 44 |                                                    | MUEARTH  | # POSITIVE VALUE                    |          |
| 45 |                                                    | R2VEC    |                                     |          |
| 46 | UNIT                                               | PDVL     | # 2D = UNIT(R2VEC)                  | (PL 8D)  |
| 47 |                                                    | R1VEC    |                                     |          |
| 48 | UNIT                                               | PUSH     | # 8D = UNIT(R1VEC)                  | (PL 14D) |
| 49 | VXV                                                | VCOMP    | # -N = UNIT(R2VEC) X UNIT(R1VEC)    |          |
| 50 |                                                    | 2D       |                                     |          |
| 51 | PUSH                                               |          | #                                   | (PL 20D) |
| 52 | LXA,1                                              | DLOAD    |                                     |          |
| 53 |                                                    |          |                                     |          |
| 54 |                                                    |          |                                     |          |
| 55 |                                                    |          |                                     |          |
| 56 |                                                    |          |                                     |          |
| 57 |                                                    |          |                                     |          |
| 58 |                                                    |          |                                     |          |
| 59 |                                                    |          |                                     |          |
| 60 |                                                    |          |                                     |          |

```
1
2 RTX1
3 18D
4 BMN INCR,1
5 +2
6 DEC -8
7 INCR,1 SLOAD
8 10D
9 X1
10 BHIZ VLOAD # (PL 14D)
11 +2
12 VCOMP PUSH # (PL 20D)
13 VLOAD # (PL 14D)
14 VXV DOT # (PL 2D)
15 BPL DLOAD # (PL 0D)
16 INITVEL4 DCOMP # (PL 2D)
17 LXA,2 SXA,2
18 OD
19 GEOMSGN
20
21 # SET INPUTS UP FOR LAMBERT
22
23 LXA,1 CALL
24 RTX1
25
26 # OPERATE THE LAMBERT CONIC ROUTINE (COASTFLT SUBROUTINE)
27
28 SETITCTR # GO TO END OF BANK TO SET ITERCTR BEFORE
29 # CALLING LAMBER (FOR REMANUFACTURE ONLY)
30
31 # ARRIVED AT SOLUTION IS GOOD ENOUGH ACCORDING TO SLIGHTLY WIDER BOUNDS.
32
33 CLEAR VLOAD
34 GUESSW
35 VVEC
36
37 # STORE CALCULATED INITIAL VELOCITY REQUIRED IN VIPRIME
38
39 STODL VIPRIME # INITIAL VELOCITY REQUIRED (+7)
40
41 # IF NUMIT IS ZERO, CONTINUE AT INITVELB, OTHERWISE
42 # SET UP INPUTS FOR ENCKE INTEGRATION (INTEGRVS).
43
44 VTARGETAG
45 BHIZ CALL
46 INITVEL7
47 INTSTALL
48 SLOAD CLEAR
49 RTX2
50 MOONFLAG
51 BHIZ SET
52 INITVEL5
```

|    |                                                                       |        |                                          |    |
|----|-----------------------------------------------------------------------|--------|------------------------------------------|----|
| 1  |                                                                       |        |                                          | 1  |
| 2  |                                                                       |        |                                          | 2  |
| 3  | INITVEL5                                                              | VLOAD  | MOONFLAG                                 | 3  |
| 4  |                                                                       |        |                                          | 4  |
| 5  |                                                                       | STORE  | RINIT                                    | 5  |
| 6  |                                                                       | STOVL  | R1VEC                                    | 6  |
| 7  |                                                                       |        |                                          | 7  |
| 8  |                                                                       | STODL  | RCV                                      | 8  |
| 9  |                                                                       |        |                                          | 9  |
| 10 |                                                                       |        | VIPRIME                                  | 10 |
| 11 |                                                                       |        | VCV                                      | 11 |
| 12 |                                                                       |        | INTIME                                   | 12 |
| 13 |                                                                       | STORE  | TET                                      | 13 |
| 14 |                                                                       | DAD    | CLEAR                                    | 14 |
| 15 |                                                                       |        |                                          | 15 |
| 16 |                                                                       |        | DELLT4                                   | 16 |
| 17 |                                                                       |        |                                          | 17 |
| 18 |                                                                       | STCALL | INTYPFLG                                 | 18 |
| 19 |                                                                       |        |                                          | 19 |
| 20 |                                                                       |        | TDEC1                                    | 20 |
| 21 |                                                                       |        | INTEGRVS                                 | 21 |
| 22 |                                                                       |        |                                          | 22 |
| 23 |                                                                       | VLOAD  |                                          | 23 |
| 24 |                                                                       |        | VATT1                                    | 24 |
| 25 |                                                                       | STORE  | VTARGET                                  | 25 |
| 26 |                                                                       |        |                                          | 26 |
| 27 | # IF ITERATION COUNTER (ITCTR) EQ NO. ITERATIONS (NUMIT), CONTINUE AT |        |                                          | 27 |
| 28 | # INITVELC, OTHERWISE REITERATE LAMBERT AND ENCKE                     |        |                                          | 28 |
| 29 |                                                                       |        |                                          | 29 |
| 30 |                                                                       | LXA,2  | INCR,2                                   | 30 |
| 31 |                                                                       |        | ITCTR                                    | 31 |
| 32 |                                                                       |        |                                          | 32 |
| 33 |                                                                       |        | ID                                       | 33 |
| 34 |                                                                       |        | # INCREMENT ITCTR                        | 34 |
| 35 |                                                                       | SXA,2  | XSU,2                                    | 35 |
| 36 |                                                                       |        | ITCTR                                    | 36 |
| 37 |                                                                       |        |                                          | 37 |
| 38 |                                                                       |        | VTARGETAG                                | 38 |
| 39 |                                                                       | SLOAD  | BHIZ                                     | 39 |
| 40 |                                                                       |        | # IF SP(MPAC) EQ 0, CONTINUE AT INITVELC | 40 |
| 41 |                                                                       |        | X2                                       | 41 |
| 42 |                                                                       |        |                                          | 42 |
| 43 |                                                                       |        | INITVEL6                                 | 43 |
| 44 |                                                                       |        |                                          | 44 |
| 45 | # OFFSET CONIC TARGET VECTOR                                          |        |                                          | 45 |
| 46 |                                                                       |        |                                          | 46 |
| 47 |                                                                       | VLOAD  | VSU                                      | 47 |
| 48 |                                                                       |        | RTARG1                                   | 48 |
| 49 |                                                                       |        |                                          | 49 |
| 50 |                                                                       | VAD    | RATT1                                    | 50 |
| 51 |                                                                       |        |                                          | 51 |
| 52 |                                                                       |        | R2VEC                                    | 52 |
| 53 |                                                                       | STODL  | R2VEC                                    | 53 |
| 54 |                                                                       |        | COZY4                                    | 54 |
| 55 |                                                                       | GOTO   |                                          | 55 |
| 56 |                                                                       |        |                                          | 56 |
| 57 |                                                                       |        | INITVEL2                                 | 57 |
| 58 |                                                                       |        | # CONTINUE ITERATING AT INITVEL2         | 58 |
| 59 |                                                                       |        |                                          | 59 |
| 60 | # COMPUTE THE DELTA VELOCITY                                          |        |                                          | 60 |
| 61 |                                                                       |        |                                          | 61 |
| 62 | INITVEL6                                                              | VLOAD  |                                          | 62 |
| 63 |                                                                       |        |                                          | 63 |
| 64 |                                                                       |        | R2VEC                                    | 64 |
| 65 |                                                                       | STORE  | RTARG1                                   | 65 |
| 66 | INITVEL7                                                              | VLOAD  | VSU                                      | 66 |
| 67 |                                                                       |        |                                          | 67 |
| 68 |                                                                       |        | VIPRIME                                  | 68 |
| 69 |                                                                       |        |                                          | 69 |
| 70 |                                                                       |        | VINIT                                    | 70 |
| 71 |                                                                       | STOVL  | DELVEET3                                 | 71 |
| 72 |                                                                       |        | # DELVEET3 = VIPRIME-VINIT (+7)          | 72 |
| 73 |                                                                       |        |                                          | 73 |
| 74 |                                                                       |        |                                          | 74 |
| 75 |                                                                       |        |                                          | 75 |
| 76 |                                                                       |        |                                          | 76 |
| 77 |                                                                       |        |                                          | 77 |
| 78 |                                                                       |        |                                          | 78 |
| 79 |                                                                       |        |                                          | 79 |
| 80 |                                                                       |        |                                          | 80 |

|    |          |                              |
|----|----------|------------------------------|
| 1  |          |                              |
| 2  |          | VTARGET                      |
| 3  |          | VTPRIME                      |
| 4  |          | BHIZ                         |
| 5  |          | RTX2                         |
| 6  |          | INITVELX                     |
| 7  |          | VSR2                         |
| 8  |          | VTPRIME                      |
| 9  |          | VTPRIME                      |
| 10 |          | VIPRIME                      |
| 11 |          | VSR2                         |
| 12 |          | STOVL                        |
| 13 |          | VIPRIME                      |
| 14 |          | RTARG1                       |
| 15 |          | RTARG1                       |
| 16 |          | DELVEET3                     |
| 17 |          | VSR2                         |
| 18 |          | STORE                        |
| 19 | INITVELX | LXA,1                        |
| 20 |          | DLOAD*                       |
| 21 |          | RTX1                         |
| 22 |          | MUTABLE -2,1                 |
| 23 |          | PUSH                         |
| 24 |          | DMP                          |
| 25 |          | R1A                          |
| 26 |          | DDV                          |
| 27 |          | R1                           |
| 28 |          | STODL                        |
| 29 |          | MU/A                         |
| 30 |          | SR                           |
| 31 |          | 6                            |
| 32 |          | MUASTEER                     |
| 33 |          | VLOAD                        |
| 34 |          | OD                           |
| 35 |          | RTARG1                       |
| 36 |          | RTARG                        |
| 37 |          | NORMEX                       |
| 38 | # *****  | END OF INITVEL ROUTINE ***** |

```
***** END OF INITVEL ROUTINE *****
```



# \*\*\*\*\* MIDGIM \*\*\*\*\*

# MOD NO. 0, BY WILLMAN, SUBROUTINE RENDGUID, LOG P34-P35, P74-P75  
# REVISION 03, 17 FEB 67

# IF THE ACTIVE VEHICLE IS DOING THE COMPUTATION, MIDGIM COMPUTES  
# THE POSITIVE MIDDLE GIMBAL ANGLE OF THE ACTIVE VEHICLE TO THE INPUT  
# DELTA VELOCITY VECTOR (OD IN PUSH LIST), OTHERWISE  
# MIDGIM CONVERTS THE INPUT DELTA VELOCITY VECTOR FROM INERTIAL COORDIN-  
# ATES TO LOCAL VERTICAL COORDINATES OF THE ACTIVE VEHICLE.

# \*\* INPUTS \*\*

| # | NAME   | MEANING                                      | UNITS/SCALING/MODE  |
|---|--------|----------------------------------------------|---------------------|
| # | AVFLAG | INT FLAG -- 0 IS CSM ACTIVE, 1 IS LEM ACTIVE | BIT                 |
| # | RINIT  | ACTIVE VEHICLE RADIUS VECTOR                 | METERS/CSEC (+7) VT |
| # | VINIT  | ACTIVE VEHICLE VELOCITY VECTOR               | METERS/CSEC (+7) VT |
| # | OD(PL) | ACTIVE VEHICLE DELTA VELOCITY VECTOR         | METERS/CSEC (+7) VT |

# \*\* OUTPUTS \*\*

| # | NAME     | MEANING                                          | UNITS/SCALING/MODE  |
|---|----------|--------------------------------------------------|---------------------|
| # | +MGA     | + MIDDLE GIMBAL ANGLE                            | REVOLUTIONS (+0) DP |
| # | DELVLVC  | DELTA VELOCITY VECTOR IN LV COORD.               | METERS/CSEC (+7) VT |
| # | MGLVFLAG | INT FLAG: 0 IS +MGA COMPUTED, 1 IS DELVLVC COMP. | BIT                 |

# \*\* CALLING SEQUENCE \*\*

| # | L   | CALL               |
|---|-----|--------------------|
| # | L+1 | MIDGIM             |
| # | L+2 | (RETURN -- ALWAYS) |

# \*\* NO SUBROUTINES CALLED \*\*

# \*\* DEBRIS -- ERASABLE TEMPORARY USAGE \*\*

# A,Q,L, PUSH LIST, MPAC.

# \*\* ALARMS -- NONE \*\*

# MIDDLE GIMBAL ANGLE COMPUTATION.

SETLOC MIDDGIM  
BANK

COUNT\* \$\$/MIDG

HALFREV 2DEC 1 B-1

GET+MGA VLOAD UNIT # (PL 0D) V (+7) TO MPAC UNITIZE UV (+1)  
UNIT

DOT SL1 # DOT UV WITH Y(STABLE MEMBER) AND RESCALE  
REFSMMAT +6 # FROM +2 TO +1 FOR ASIN ROUTINE

ARCSIN BPL

DAD SETMGA  
DAD # CONVERT -MGA TO +MGA BY  
HALFREV # ADDING ONE REVOLUTION

SETMGA STORE +MGA  
CLR RVQ # CLEAR MGLVFLAG TO INDICATE +MGA CALC

GET.LVC VLOAD MGLVFLAG # AND EXIT  
UNIT # (PL 6D) R (+29) IN MPAC UNITIZE UR  
RINIT

VCOMP # U(-R)  
STORE 18D # U(-R) TO 18D  
VXV UNIT # U(-R)\*V EQ V\*U(R), U(V\*R)

VINIT  
STORE 12D # U(V\*R) TO 12D  
VXV UNIT # U(V\*R)\*U(-R), U((V\*R)\*(-R))

STOVL 18D  
6D # TRANSFORMATION MATRIX IS IN 6D (+1)  
0D # DELTA V (+7) IN 0D

MXV VSL1 # CONVERT FROM INER COOR TO LV COOR (+8)  
6D # AND SCALE +7 IN MPAC  
STORE DELVLVC # STORE IN DELVLVC (+7)

SET RVQ # SET MGLVFLAG TO INDICATE LVC CALC  
MGLVFLAG # AND EXIT

# \*\*\*\*\* END OF MIDGIM ROUTINE \*\*\*\*\*

|    |          |        |              |    |
|----|----------|--------|--------------|----|
| 1  |          |        |              | 1  |
| 2  |          | BANK   | 10           | 2  |
| 3  |          | SETLOC | SLCTMU       | 3  |
| 4  |          | BANK   |              | 4  |
| 5  |          | COUNT* | \$\$/MIDG    | 5  |
| 6  |          |        |              | 6  |
| 7  | SELECTMU | AXC,1  | AXT,2        | 7  |
| 8  |          |        | 2D           | 8  |
| 9  |          |        | 0D           | 9  |
| 10 |          | BOFF   |              | 10 |
| 11 |          |        | CMOONFLG     | 11 |
| 12 |          |        | SETMUER      | 12 |
| 13 |          | AXC,1  | AXT,2        | 13 |
| 14 |          |        | 10D          | 14 |
| 15 |          |        | 2D           | 15 |
| 16 | SETMUER  | DLOAD* | SXA,1        | 16 |
| 17 |          |        | MUTABLE +4,1 | 17 |
| 18 |          |        | RTX1         | 18 |
| 19 |          | STODL* | RTSR1/MU     | 19 |
| 20 |          |        | MUTABLE -2,1 | 20 |
| 21 |          | BOFF   | SR           | 21 |
| 22 |          |        | CMOONFLG     | 22 |
| 23 |          |        | RTRNMU       | 23 |
| 24 |          |        | 6D           | 24 |
| 25 | RTRNMU   | STORE  | RTMU         | 25 |
| 26 |          | SXA,2  | CLEAR        | 26 |
| 27 |          |        | RTX2         | 27 |
| 28 |          |        | FINALFLG     | 28 |
| 29 |          | GOTO   |              | 29 |
| 30 |          |        | VN1645       | 30 |
| 31 |          |        |              | 31 |
| 32 |          |        |              | 32 |
| 33 |          |        |              | 33 |
| 34 |          |        |              | 34 |
| 35 |          |        |              | 35 |
| 36 |          |        |              | 36 |
| 37 |          |        |              | 37 |
| 38 |          |        |              | 38 |
| 39 |          |        |              | 39 |
| 40 |          |        |              | 40 |
| 41 |          |        |              | 41 |
| 42 |          |        |              | 42 |
| 43 |          |        |              | 43 |
| 44 |          |        |              | 44 |
| 45 |          |        |              | 45 |
| 46 |          |        |              | 46 |
| 47 |          |        |              | 47 |
| 48 |          |        |              | 48 |
| 49 |          |        |              | 49 |
| 50 |          |        |              | 50 |
| 51 |          |        |              | 51 |
| 52 |          |        |              | 52 |
| 53 |          |        |              | 53 |
| 54 |          |        |              | 54 |
| 55 |          |        |              | 55 |
| 56 |          |        |              | 56 |
| 57 |          |        |              | 57 |
| 58 |          |        |              | 58 |
| 59 |          |        |              | 59 |
| 60 |          |        |              | 60 |

# \*\*\*\*\* PERIAPO \*\*\*\*\*

# MOD NO -1 LOG SECTION -- P34-P35, P74-P75  
# MOD BY WHITE, P. DATE 18 JAN 68

# FUNCTIONAL DESCRIPTION

# THIS SUBROUTINE COMPUTES THE TWO BODY APOCENTER AND PERICENTER  
# ALTITUDES GIVEN THE POSITION AND VELOCITY VECTORS FOR A POINT ON  
# TRAJECTORY AND THE PRIMARY BODY.

# SETRAD IS CALLED TO DETERMINE THE RADIUS OF THE PRIMARY BODY.

# APSIDES IS CALLED TO SOVE FOR THE TWO BODY RADII OF APOCENTER AND  
# PERICENTER AND THE ECCENTRICITY OF THE TRAJECTORY.

# CALLING SEQUENCE

# L CALL  
# L+1 PERIAPO  
# L+2 (RETURN -- ALWAYS)

# INPUT

# (1) RVEC POSITION VECTOR IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (2) VVEC VELOCITY VECTOR IN METERS/CENTISECOND  
# SCALE FACTOR -- EARTH +7, MOON +5  
# (3) X1 PRIMARY BODY INDICATOR  
# EARTH -1, MOON -10

# OUTPUT

# (1) 2D APOCENTER RADIUS IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (2) 4D APOCENTER ALTITUDE IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (3) 6D PERICENTER RADIUS IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (4) 8D PERICENTER ALTITUDE IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (5) ECC ECCENTRICITY OF CONIC TRAJECTORY  
# SCALE FACTOR -- +3  
# (6) XXXALT RADIUS OF THE PRIMARY BODY IN METERS  
# SCALE FACTOR -- EARTH +29, MOON +27  
# (7) PUSHLOC EQUALS 10D

# SUBROUTINES USED

# SETRAD

# P34-35\_P74-75

|    |                      |        |              |                                |            |    |
|----|----------------------|--------|--------------|--------------------------------|------------|----|
| 1  | # APSIDES            |        |              |                                |            | 1  |
| 2  |                      |        |              |                                |            | 2  |
| 3  |                      |        |              |                                |            | 3  |
| 4  |                      | SETLOC | APOPERI      |                                |            | 4  |
| 5  |                      | BANK   |              |                                |            | 5  |
| 6  |                      |        |              |                                |            | 6  |
| 7  |                      | COUNT* | \$\$/PERAP   |                                |            | 7  |
| 8  |                      |        |              |                                |            | 8  |
| 9  | RPAD                 | 2DEC   | 6373338 B-29 | # STANDARD RADIUS OF PAD 37-B. |            | 9  |
| 10 | # = 20 909 901.57 FT |        |              |                                |            | 10 |
| 11 |                      |        |              |                                |            | 11 |
| 12 | PERIAP01             | LXA,2  | VSR*         |                                |            | 12 |
| 13 |                      |        | RTX2         |                                |            | 13 |
| 14 |                      |        | 0,2          |                                |            | 14 |
| 15 |                      | STOVL  | VVEC         |                                |            | 15 |
| 16 |                      |        | LXA,1        | VSR*                           |            | 16 |
| 17 |                      |        | RTX1         |                                |            | 17 |
| 18 |                      |        | 0,2          |                                |            | 18 |
| 19 | PERIAPO              | STORE  | RVEC         |                                |            | 19 |
| 20 |                      | STQ    | CALL         |                                |            | 20 |
| 21 |                      |        | NORMEX       |                                |            | 21 |
| 22 |                      |        | SETRAD       |                                |            | 22 |
| 23 |                      |        | XXXALT       |                                |            | 23 |
| 24 |                      |        | APSIDES      |                                |            | 24 |
| 25 |                      | SETPD  | PUSH         | # 2D = APOCENTER RADIUS        | B29 OR B27 | 25 |
| 26 |                      |        | 2D           |                                |            | 26 |
| 27 |                      | DSU    | PDDL         | # 4D = APOGEE ALTITUDE         | B29 OR B27 | 27 |
| 28 |                      |        | XXXALT       |                                |            | 28 |
| 29 |                      |        | OD           |                                |            | 29 |
| 30 |                      | PUSH   | DSU          | # 6D = PERICENTER RADIUS       | B29 OR B27 | 30 |
| 31 |                      |        | XXXALT       |                                |            | 31 |
| 32 |                      | PUSH   | GOTO         | # 8D = PERIGEE ALTITUDE        | B29 OR B27 | 32 |
| 33 |                      |        | NORMEX       |                                |            | 33 |
| 34 |                      |        |              |                                |            | 34 |
| 35 |                      |        |              |                                |            | 35 |
| 36 |                      |        |              |                                |            | 36 |
| 37 |                      |        |              |                                |            | 37 |
| 38 |                      |        |              |                                |            | 38 |
| 39 |                      |        |              |                                |            | 39 |
| 40 |                      |        |              |                                |            | 40 |
| 41 |                      |        |              |                                |            | 41 |
| 42 |                      |        |              |                                |            | 42 |
| 43 |                      |        |              |                                |            | 43 |
| 44 |                      |        |              |                                |            | 44 |
| 45 |                      |        |              |                                |            | 45 |
| 46 |                      |        |              |                                |            | 46 |
| 47 |                      |        |              |                                |            | 47 |
| 48 |                      |        |              |                                |            | 48 |
| 49 |                      |        |              |                                |            | 49 |
| 50 |                      |        |              |                                |            | 50 |
| 51 |                      |        |              |                                |            | 51 |
| 52 |                      |        |              |                                |            | 52 |
| 53 |                      |        |              |                                |            | 53 |
| 54 |                      |        |              |                                |            | 54 |
| 55 |                      |        |              |                                |            | 55 |
| 56 |                      |        |              |                                |            | 56 |
| 57 |                      |        |              |                                |            | 57 |
| 58 |                      |        |              |                                |            | 58 |
| 59 |                      |        |              |                                |            | 59 |
| 60 |                      |        |              |                                |            | 60 |



|    |         |       |         |    |
|----|---------|-------|---------|----|
| 1  |         |       |         | 1  |
| 2  | SETRAD  | DLOAD | PUSH    | 2  |
| 3  |         |       | RPAD    | 3  |
| 4  |         | SXA,1 | INCR,2  | 4  |
| 5  |         |       | X2      | 5  |
| 6  |         |       | 2D      | 6  |
| 7  |         | SLOAD | BHIZ    | 7  |
| 8  |         |       | X2      | 8  |
| 9  |         |       | SETRADX | 9  |
| 10 |         | VLOAD | ABVAL   | 10 |
| 11 |         |       | RLS     | 11 |
| 12 |         | PDDL  |         | 12 |
| 13 | SETRADX | DLOAD | RVQ     | 13 |
| 14 |         |       |         | 14 |
| 15 |         |       |         | 15 |
| 16 |         |       |         | 16 |
| 17 |         |       |         | 17 |
| 18 |         |       |         | 18 |
| 19 |         |       |         | 19 |
| 20 |         |       |         | 20 |
| 21 |         |       |         | 21 |
| 22 |         |       |         | 22 |
| 23 |         |       |         | 23 |
| 24 |         |       |         | 24 |
| 25 |         |       |         | 25 |
| 26 |         |       |         | 26 |
| 27 |         |       |         | 27 |
| 28 |         |       |         | 28 |
| 29 |         |       |         | 29 |
| 30 |         |       |         | 30 |
| 31 |         |       |         | 31 |
| 32 |         |       |         | 32 |
| 33 |         |       |         | 33 |
| 34 |         |       |         | 34 |
| 35 |         |       |         | 35 |
| 36 |         |       |         | 36 |
| 37 |         |       |         | 37 |
| 38 |         |       |         | 38 |
| 39 |         |       |         | 39 |
| 40 |         |       |         | 40 |
| 41 |         |       |         | 41 |
| 42 |         |       |         | 42 |
| 43 |         |       |         | 43 |
| 44 |         |       |         | 44 |
| 45 |         |       |         | 45 |
| 46 |         |       |         | 46 |
| 47 |         |       |         | 47 |
| 48 |         |       |         | 48 |
| 49 |         |       |         | 49 |
| 50 |         |       |         | 50 |
| 51 |         |       |         | 51 |
| 52 |         |       |         | 52 |
| 53 |         |       |         | 53 |
| 54 |         |       |         | 54 |
| 55 |         |       |         | 55 |
| 56 |         |       |         | 56 |
| 57 |         |       |         | 57 |
| 58 |         |       |         | 58 |
| 59 |         |       |         | 59 |
| 60 |         |       |         | 60 |

|    |          |        |          |    |
|----|----------|--------|----------|----|
| 1  |          |        |          | 1  |
| 2  | PRECSET  | STQ    |          | 2  |
| 3  |          |        | NORMEX   | 3  |
| 4  |          | STCALL | TDEC2    | 4  |
| 5  |          |        | LEMPREC  | 5  |
| 6  |          | CALL   |          | 6  |
| 7  |          |        | LEMSTORE | 7  |
| 8  |          | DLOAD  |          | 8  |
| 9  |          |        | TDEC2    | 9  |
| 10 |          | STCALL | TDEC1    | 10 |
| 11 |          |        | CSMPREC  | 11 |
| 12 |          | CALL   |          | 12 |
| 13 |          |        | CSMSTORE | 13 |
| 14 |          | GOTO   |          | 14 |
| 15 |          |        | NORMEX   | 15 |
| 16 | LEMSTORE | VLOAD  | BOFF     | 16 |
| 17 |          |        | RATT     | 17 |
| 18 |          |        | AVFLAG   | 18 |
| 19 | ACTIVE   |        | PASSIVE  | 19 |
| 20 |          | STOVL  | RACT3    | 20 |
| 21 |          |        | VATT     | 21 |
| 22 |          | STORE  | VACT3    | 22 |
| 23 | CSMSTORE | RVQ    |          | 23 |
| 24 |          | VLOAD  | BOFF     | 24 |
| 25 |          |        | RATT     | 25 |
| 26 |          |        | AVFLAG   | 26 |
| 27 |          |        | ACTIVE   | 27 |
| 28 | PASSIVE  | STOVL  | RPASS3   | 28 |
| 29 |          |        | VATT     | 29 |
| 30 |          | STORE  | VPASS3   | 30 |
| 31 |          | RVQ    |          | 31 |
| 32 |          |        |          | 32 |
| 33 |          |        |          | 33 |
| 34 |          |        |          | 34 |
| 35 |          |        |          | 35 |
| 36 |          |        |          | 36 |
| 37 |          |        |          | 37 |
| 38 |          |        |          | 38 |
| 39 |          |        |          | 39 |
| 40 |          |        |          | 40 |
| 41 |          |        |          | 41 |
| 42 |          |        |          | 42 |
| 43 |          |        |          | 43 |
| 44 |          |        |          | 44 |
| 45 |          |        |          | 45 |
| 46 |          |        |          | 46 |
| 47 |          |        |          | 47 |
| 48 |          |        |          | 48 |
| 49 |          |        |          | 49 |
| 50 |          |        |          | 50 |
| 51 |          |        |          | 51 |
| 52 |          |        |          | 52 |
| 53 |          |        |          | 53 |
| 54 |          |        |          | 54 |
| 55 |          |        |          | 55 |
| 56 |          |        |          | 56 |
| 57 |          |        |          | 57 |
| 58 |          |        |          | 58 |
| 59 |          |        |          | 59 |
| 60 |          |        |          | 60 |



|    |          |       |      |    |
|----|----------|-------|------|----|
| 1  |          |       |      | 1  |
| 2  | VECSHIFT | LXA,2 | VSR* | 2  |
| 3  |          |       | RTX2 | 3  |
| 4  |          |       | 0,2  | 4  |
| 5  |          | LXA,1 | PDVL | 5  |
| 6  |          |       | RTX1 | 6  |
| 7  |          | VSR*  | PDVL | 7  |
| 8  |          |       | 0,2  | 8  |
| 9  |          | RVQ   |      | 9  |
| 10 |          |       |      | 10 |
| 11 |          |       |      | 11 |
| 12 |          |       |      | 12 |
| 13 |          |       |      | 13 |
| 14 |          |       |      | 14 |
| 15 |          |       |      | 15 |
| 16 |          |       |      | 16 |
| 17 |          |       |      | 17 |
| 18 |          |       |      | 18 |
| 19 |          |       |      | 19 |
| 20 |          |       |      | 20 |
| 21 |          |       |      | 21 |
| 22 |          |       |      | 22 |
| 23 |          |       |      | 23 |
| 24 |          |       |      | 24 |
| 25 |          |       |      | 25 |
| 26 |          |       |      | 26 |
| 27 |          |       |      | 27 |
| 28 |          |       |      | 28 |
| 29 |          |       |      | 29 |
| 30 |          |       |      | 30 |
| 31 |          |       |      | 31 |
| 32 |          |       |      | 32 |
| 33 |          |       |      | 33 |
| 34 |          |       |      | 34 |
| 35 |          |       |      | 35 |
| 36 |          |       |      | 36 |
| 37 |          |       |      | 37 |
| 38 |          |       |      | 38 |
| 39 |          |       |      | 39 |
| 40 |          |       |      | 40 |
| 41 |          |       |      | 41 |
| 42 |          |       |      | 42 |
| 43 |          |       |      | 43 |
| 44 |          |       |      | 44 |
| 45 |          |       |      | 45 |
| 46 |          |       |      | 46 |
| 47 |          |       |      | 47 |
| 48 |          |       |      | 48 |
| 49 |          |       |      | 49 |
| 50 |          |       |      | 50 |
| 51 |          |       |      | 51 |
| 52 |          |       |      | 52 |
| 53 |          |       |      | 53 |
| 54 |          |       |      | 54 |
| 55 |          |       |      | 55 |
| 56 |          |       |      | 56 |
| 57 |          |       |      | 57 |
| 58 |          |       |      | 58 |
| 59 |          |       |      | 59 |
| 60 |          |       |      | 60 |





|    |         |       |      |    |
|----|---------|-------|------|----|
| 1  |         |       |      | 1  |
| 2  | SHIFTR1 | LXA,2 | SL*  | 2  |
| 3  |         |       | RTX2 | 3  |
| 4  |         |       | 0,2  | 4  |
| 5  |         | RVQ   |      | 5  |
| 6  |         |       |      | 6  |
| 7  |         |       |      | 7  |
| 8  |         |       |      | 8  |
| 9  |         |       |      | 9  |
| 10 |         |       |      | 10 |
| 11 |         |       |      | 11 |
| 12 |         |       |      | 12 |
| 13 |         |       |      | 13 |
| 14 |         |       |      | 14 |
| 15 |         |       |      | 15 |
| 16 |         |       |      | 16 |
| 17 |         |       |      | 17 |
| 18 |         |       |      | 18 |
| 19 |         |       |      | 19 |
| 20 |         |       |      | 20 |
| 21 |         |       |      | 21 |
| 22 |         |       |      | 22 |
| 23 |         |       |      | 23 |
| 24 |         |       |      | 24 |
| 25 |         |       |      | 25 |
| 26 |         |       |      | 26 |
| 27 |         |       |      | 27 |
| 28 |         |       |      | 28 |
| 29 |         |       |      | 29 |
| 30 |         |       |      | 30 |
| 31 |         |       |      | 31 |
| 32 |         |       |      | 32 |
| 33 |         |       |      | 33 |
| 34 |         |       |      | 34 |
| 35 |         |       |      | 35 |
| 36 |         |       |      | 36 |
| 37 |         |       |      | 37 |
| 38 |         |       |      | 38 |
| 39 |         |       |      | 39 |
| 40 |         |       |      | 40 |
| 41 |         |       |      | 41 |
| 42 |         |       |      | 42 |
| 43 |         |       |      | 43 |
| 44 |         |       |      | 44 |
| 45 |         |       |      | 45 |
| 46 |         |       |      | 46 |
| 47 |         |       |      | 47 |
| 48 |         |       |      | 48 |
| 49 |         |       |      | 49 |
| 50 |         |       |      | 50 |
| 51 |         |       |      | 51 |
| 52 |         |       |      | 52 |
| 53 |         |       |      | 53 |
| 54 |         |       |      | 54 |
| 55 |         |       |      | 55 |
| 56 |         |       |      | 56 |
| 57 |         |       |      | 57 |
| 58 |         |       |      | 58 |
| 59 |         |       |      | 59 |
| 60 |         |       |      | 60 |

```
PROGRAM DESCRIPTION
SUBROUTINE NAME R36 OUT-OF-PLANE RENDEZVOUS ROUTINE
MOD NO. 0 DATE 22 DECEMBER 67
MOD BY N.M.NEVILLE LOG SECTION EXTENDED VERBS
FUNCTIONAL DESCRIPTION
#
TO DISPLAY AT ASTRONAUT REQUEST LGC CALCULATED RENDEZVOUS
OUT-OF-PLANE PARAMETERS (Y, YDOT, PSI). (REQUESTED BY DSKY).
#
CALLING SEQUENCE
#
ASTRONAUT REQUEST THROUGH DSKY V 90 E
#
SUBROUTINES CALLED
#
EXDSPRET
GOMARKF
CSMPREC
LEMPREC
SGNAGREE
LOADTIME
#
NORMAL EXIT MODES
#
ASTRONAUT REQUEST THROUGH DSKY TO TERMINATE PROGRAM V 34 E
#
ALARM OR ABORT EXIT MODES
#
NONE
#
OUTPUT
#
DECIMAL DISPLAY OF TIME, Y, YDOT AND PSI
#
DISPLAYED VALUES Y, YDOT, AND PSI, ARE STORED IN ERASABLE
REGISTERS RANGE, RRATE, AND RTHETA RESPECTIVELY.
#
ERASABLE INITIALIZATION REQUIRED
#
CSM AND LEM STATE VECTORS
#
DEBRIS
#
CENTRALS A,Q,L
OTHER: THOSE USED BY THE ABOVE LISTED SUBROUTINES
#
BANK 20
SETLOC R36LM
BANK
```

|    |          |        |          |                                                  |    |
|----|----------|--------|----------|--------------------------------------------------|----|
| 1  |          |        |          |                                                  | 1  |
| 2  |          | EBANK= | RPASS36  |                                                  | 2  |
| 3  |          |        |          |                                                  | 3  |
| 4  |          | COUNT* | \$\$/R36 |                                                  | 4  |
| 5  |          |        |          |                                                  | 5  |
| 6  | R36      | ZL     |          |                                                  | 6  |
| 7  |          | CAF    | ZERO     | # SET TIME OF EVENT TO ZERO FOR FIRST            | 7  |
| 8  |          | DXCH   | DSPTMX   | # DISPLAY                                        | 8  |
| 9  |          | CAF    | V06N16N  |                                                  | 9  |
| 10 |          | TC     | BANKCALL |                                                  | 10 |
| 11 |          | CADR   | GOMARKF  |                                                  | 11 |
| 12 |          | TCF    | ENDEXT   | # TERMINATE                                      | 12 |
| 13 |          | TCF    | +2       | # PROCEED                                        | 13 |
| 14 |          | TCF    | -5       | # RECYCLE FOR ASTRONAUT INPUT TIME               | 14 |
| 15 |          | DXCH   | DSPTMX   |                                                  | 15 |
| 16 |          | EXTEND |          |                                                  | 16 |
| 17 |          | BZF    | LREGCHK  | # A-REG ZERO GOTO CHECK L-REG FOR ZERO           | 17 |
| 18 | ASTROTIM | DXCH   | MPAC     | # A-REG NON-ZERO, TIME = ASTRO INPUT TIME        | 18 |
| 19 |          | TC     | INTPRET  |                                                  | 19 |
| 20 |          | RTB    |          |                                                  | 20 |
| 21 |          |        | DPMODE   |                                                  | 21 |
| 22 | R36INT   | STCALL | TDEC1    |                                                  | 22 |
| 23 |          |        | OTHPREC  |                                                  | 23 |
| 24 |          | VLOAD  | PDVL     |                                                  | 24 |
| 25 |          |        | VATT     |                                                  | 25 |
| 26 |          |        | RATT     | #                                                | 26 |
| 27 |          | STORE  | RPASS36  | # $\bar{R}$                                      | 27 |
| 28 |          | UNIT   | PDVL     | # P                                              | 28 |
| 29 |          | VXV    | UNIT     |                                                  | 29 |
| 30 |          | STADR  |          | #                                                | 30 |
| 31 |          | STODL  | UNP36    | # $\bar{U}$                                      | 31 |
| 32 |          |        | TAT      |                                                  | 32 |
| 33 |          | STCALL | TDEC1    |                                                  | 33 |
| 34 |          |        | THISPREC |                                                  | 34 |
| 35 |          | VLOAD  | PDVL     | #                                                | 35 |
| 36 |          |        | VATT     | # VELOCITY VECTOR $\bar{V}$ 00D                  | 36 |
| 37 |          |        | RATT     | # A                                              | 37 |
| 38 |          | PDDL   |          |                                                  | 38 |
| 39 |          |        | TAT      | # SAVE TIME IN LOCATION 30D FOR REDISPLAY        | 39 |
| 40 |          | STOVL  | 30D      | #                                                | 40 |
| 41 |          | PUSH   | PUSH     | # POSITION VECTOR $\bar{R}$ IN 06D AND 12D       | 41 |
| 42 |          | BVSU   | PDVL     | # A                                              | 42 |
| 43 |          |        | RPASS36  | # LINE OF SIGHT VECTOR $\bar{R} - \bar{R}_P$ 12D | 43 |
| 44 |          | DOT    | SL1      | #                                                | 44 |
| 45 |          |        | UNP36    | #                                                | 45 |
| 46 |          | STOVL  | RANGE    | # $\bar{Y} = \bar{U} \cdot \bar{R}_A$            | 46 |
| 47 |          |        | 00D      | #                                                | 47 |
| 48 |          | DOT    | SL1      |                                                  | 48 |
| 49 |          |        | UNP36    | # $\dot{\bar{Y}} = \bar{U} \cdot \bar{V}_A$      | 49 |
| 50 |          | STOVL  | RRATE    | #                                                | 50 |
| 51 |          |        | 06D      | #                                                | 51 |
| 52 |          |        |          |                                                  | 52 |
| 53 |          |        |          |                                                  | 53 |
| 54 |          |        |          |                                                  | 54 |
| 55 |          |        |          |                                                  | 55 |
| 56 |          |        |          |                                                  | 56 |
| 57 |          |        |          |                                                  | 57 |
| 58 |          |        |          |                                                  | 58 |
| 59 |          |        |          |                                                  | 59 |
| 60 |          |        |          |                                                  | 60 |

|    |         |        |          |                                                         |    |
|----|---------|--------|----------|---------------------------------------------------------|----|
| 1  |         |        |          |                                                         | 1  |
| 2  |         |        |          |                                                         | 2  |
| 3  |         | UNIT   | PUSH     | # $\bar{U} = \text{UNIT} ( \bar{R} )$ 18D               | 3  |
| 4  |         | VXV    | VXV      | # RA A                                                  | 4  |
| 5  |         |        | 00D      | #                                                       | 5  |
| 6  |         |        | 18D      | # $( \bar{U} \times \bar{V} ) \times \bar{U} = \bar{U}$ | 6  |
| 7  |         | VSL2   | UNIT     | # RA A RA A                                             | 7  |
| 8  |         | UNIT   |          |                                                         | 8  |
| 9  |         | STOVL  | 00D      | # UNIT HORIZONTAL IN FORWARD DIR. 00D                   | 9  |
| 10 |         |        | 18D      |                                                         | 10 |
| 11 |         | DOT    | VXSC     | # $\bar{U}$                                             | 11 |
| 12 |         |        | 12D      | # L                                                     | 12 |
| 13 |         | VSL2   |          |                                                         | 13 |
| 14 |         | BVSU   | UNIT     |                                                         | 14 |
| 15 |         | UNIT   |          |                                                         | 15 |
| 16 |         | PUSH   | DOT      | # LOS PROJECTED INTO HORIZONTAL 12D                     | 16 |
| 17 |         |        | 00D      | # PLANE                                                 | 17 |
| 18 |         | SL1    | ARCCOS   | #                                                       | 18 |
| 19 |         | STOVL  | RTHETA   | # $\text{PSI} = \text{ARCCOS}( \bar{U} \cdot \bar{U} )$ | 19 |
| 20 |         | VXV    | DOT      | # A L                                                   | 20 |
| 21 |         |        | 00D      |                                                         | 21 |
| 22 |         | BPL    | DLOAD    |                                                         | 22 |
| 23 |         |        | R36TAG2  |                                                         | 23 |
| 24 |         |        | LODPMAX  |                                                         | 24 |
| 25 |         | DSU    |          |                                                         | 25 |
| 26 |         |        | RTHETA   |                                                         | 26 |
| 27 |         | STORE  | RTHETA   |                                                         | 27 |
| 28 | R36TAG2 | DLOAD  | RTB      |                                                         | 28 |
| 29 |         |        | 30D      |                                                         | 29 |
| 30 |         |        | SGNAGREE |                                                         | 30 |
| 31 |         | STORE  | DSPTMX   |                                                         | 31 |
| 32 |         | EXIT   |          |                                                         | 32 |
| 33 |         | CAF    | V06N90N  | # DISPLAY Y, YDOT, AND PSI.                             | 33 |
| 34 |         | TC     | BANKCALL |                                                         | 34 |
| 35 |         | CADR   | GOMARKF  |                                                         | 35 |
| 36 |         | TCF    | ENDEXT   | # TERMINATE                                             | 36 |
| 37 |         | TCF    | ENDEXT   | # PROCEED, END OF PROGRAM                               | 37 |
| 38 |         | TCF    | R36 +3   | # REDISPLAY OUTPUT                                      | 38 |
| 39 | LREGCHK | XCH    | L        |                                                         | 39 |
| 40 |         | EXTEND |          |                                                         | 40 |
| 41 |         | BZF    | ENTTIM2  | # L-REG ZERO, SET TIME = PRESENT TIME                   | 41 |
| 42 |         | XCH    | L        | # L-REG NON ZERO, TIME = ASTRO INPUT TIME               | 42 |
| 43 |         | TCF    | ASTROTIM |                                                         | 43 |
| 44 | ENTTIM2 | TC     | INTPRET  |                                                         | 44 |
| 45 |         | RTB    | GOTO     |                                                         | 45 |
| 46 |         |        | LOADTIME |                                                         | 46 |
| 47 |         |        | R36INT   |                                                         | 47 |
| 48 | V06N16N | VN     | 00616    |                                                         | 48 |
| 49 | V06N90N | VN     | 00690    |                                                         | 49 |
| 50 |         |        |          |                                                         | 50 |
| 51 |         |        |          |                                                         | 51 |
| 52 |         |        |          |                                                         | 52 |
| 53 |         |        |          |                                                         | 53 |
| 54 |         |        |          |                                                         | 54 |
| 55 |         |        |          |                                                         | 55 |
| 56 |         |        |          |                                                         | 56 |
| 57 |         |        |          |                                                         | 57 |
| 58 |         |        |          |                                                         | 58 |
| 59 |         |        |          |                                                         | 59 |
| 60 |         |        |          |                                                         | 60 |

|    |          |        |          |    |
|----|----------|--------|----------|----|
| 1  | # R31    |        |          | 1  |
| 2  |          | BANK   | 40       | 2  |
| 3  |          | SETLOC | R31LOC   | 3  |
| 4  |          | BANK   |          | 4  |
| 5  |          |        |          | 5  |
| 6  |          | COUNT* | \$\$/R31 | 6  |
| 7  |          |        |          | 7  |
| 8  | R31CALL  | CAF    | PRI03    | 8  |
| 9  |          | TC     | FINDVAC  | 9  |
| 10 |          | EBANK= | SUBEXIT  | 10 |
| 11 |          | 2CADR  | V83CALL  | 11 |
| 12 |          |        |          | 12 |
| 13 | DSPDELAY | TC     | FIXDELAY | 13 |
| 14 |          | DEC    | 100      | 14 |
| 15 |          | CA     | EXTVBACT | 15 |
| 16 |          | MASK   | BIT12    | 16 |
| 17 |          | EXTEND |          | 17 |
| 18 |          | BZF    | DSPDELAY | 18 |
| 19 |          |        |          | 19 |
| 20 |          | CAF    | PRI05    | 20 |
| 21 |          | TC     | NOVAC    | 21 |
| 22 |          | EBANK= | TSTRT    | 22 |
| 23 |          | 2CADR  | DISPN5X  | 23 |
| 24 |          |        |          | 24 |
| 25 |          | TCF    | TASKOVER | 25 |
| 26 |          |        |          | 26 |
| 27 |          | BANK   | 37       | 27 |
| 28 |          | SETLOC | R31      | 28 |
| 29 |          | BANK   |          | 29 |
| 30 |          | COUNT* | \$\$/R31 | 30 |
| 31 |          |        |          | 31 |
| 32 | DISPN5X  | CAF    | V16N54   | 32 |
| 33 |          | TC     | BANKCALL | 33 |
| 34 |          | CADR   | GOMARKF  | 34 |
| 35 |          | TC     | B50FF    | 35 |
| 36 |          | TC     | B50FF    | 36 |
| 37 |          | TCF    | DISPN5X  | 37 |
| 38 |          |        |          | 38 |
| 39 | V83CALL  | CS     | FLAGWRD7 | 39 |
| 40 |          | MASK   | AVEGFBIT | 40 |
| 41 |          | EXTEND |          | 41 |
| 42 |          | BZF    | MUNG?    | 42 |
| 43 |          |        |          | 43 |
| 44 |          | CS     | FLAGWRD8 | 44 |
| 45 |          | MASK   | SURFFBIT | 45 |
| 46 |          | EXTEND |          | 46 |
| 47 |          | BZF    | ONEBASE  | 47 |
| 48 |          |        |          | 48 |
| 49 |          | TC     | INTPRET  | 49 |
| 50 |          | RTB    |          | 50 |
| 51 |          |        |          | 51 |
| 52 |          |        |          | 52 |
| 53 |          |        |          | 53 |
| 54 |          |        |          | 54 |
| 55 |          |        |          | 55 |
| 56 |          |        |          | 56 |
| 57 |          |        |          | 57 |
| 58 |          |        |          | 58 |
| 59 |          |        |          | 59 |
| 60 |          |        |          | 60 |

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  | # RS1    |        | PAGE 101 |                                            | 1  |
| 2  |          |        | LOADTIME |                                            | 2  |
| 3  |          | STCALL | TDEC1    |                                            | 3  |
| 4  |          |        | LEMPREC  | # PRECISION BASE VECTOR FOR LM             | 4  |
| 5  |          | VLOAD  |          |                                            | 5  |
| 6  |          |        | RATT1    |                                            | 6  |
| 7  |          | STOVL  | BASETHP  |                                            | 7  |
| 8  |          |        | VATT1    |                                            | 8  |
| 9  |          | STODL  | BASETHV  |                                            | 9  |
| 10 |          |        | TAT      |                                            | 10 |
| 11 | DOCMBASE | STORE  | BASETIME | # PRECISION BASE VECTOR FOR CM             | 11 |
| 12 |          | STCALL | TDEC1    |                                            | 12 |
| 13 |          |        | CSMPREC  |                                            | 13 |
| 14 |          | VLOAD  |          |                                            | 14 |
| 15 |          |        | RATT1    |                                            | 15 |
| 16 |          | STOVL  | BASEOTP  |                                            | 16 |
| 17 |          |        | VATT1    |                                            | 17 |
| 18 |          | STORE  | BASEOTV  |                                            | 18 |
| 19 |          | EXIT   |          |                                            | 19 |
| 20 |          |        |          |                                            | 20 |
| 21 | REV83    | CS     | FLAGWRD7 |                                            | 21 |
| 22 |          | MASK   | AVEGFBIT |                                            | 22 |
| 23 |          | EXTEND |          |                                            | 23 |
| 24 |          | BZF    | GETRVN   | # IF AVEGFLAG SET, USE RN,VN               | 24 |
| 25 |          |        |          |                                            | 25 |
| 26 |          | CS     | FLAGWRD8 |                                            | 26 |
| 27 |          | MASK   | SURFFBIT |                                            | 27 |
| 28 |          | EXTEND |          |                                            | 28 |
| 29 |          | BZF    | R31SURF  | # IF ON SURFACE,USE LEMAREC                | 29 |
| 30 |          |        |          |                                            | 30 |
| 31 |          | TC     | INTPRET  | # DO CONIC EXTRAPOLATION FOR BOTH VEHICLES | 31 |
| 32 |          | RTB    |          |                                            | 32 |
| 33 |          |        | LOADTIME |                                            | 33 |
| 34 |          | STCALL | TDEC1    |                                            | 34 |
| 35 |          |        | INTSTALL |                                            | 35 |
| 36 |          | VLOAD  | CLEAR    |                                            | 36 |
| 37 |          |        | BASETHP  |                                            | 37 |
| 38 |          |        | MOONFLAG |                                            | 38 |
| 39 |          | STOVL  | RCV      |                                            | 39 |
| 40 |          |        | BASETHV  |                                            | 40 |
| 41 |          | STODL  | VCV      |                                            | 41 |
| 42 |          |        | BASETIME |                                            | 42 |
| 43 |          | BOF    | SET      | # GET APPROPRIATE MOONFLAG SETTING         | 43 |
| 44 |          |        | MOONTHIS |                                            | 44 |
| 45 |          |        | +2       |                                            | 45 |
| 46 |          |        | MOONFLAG |                                            | 46 |
| 47 |          | SET    |          |                                            | 47 |
| 48 |          |        | INTYPFLG | # CONIC EXTRAP.                            | 48 |
| 49 |          | STCALL | TET      |                                            | 49 |
| 50 |          |        | INTEGRVS | # INTEGRATION --- AT LAST---               | 50 |
| 51 | OTHCONIC | VLOAD  |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

|    |          |        |          |                                    |    |
|----|----------|--------|----------|------------------------------------|----|
| 1  | # RSI    |        |          | PAGE 103                           | 1  |
| 2  |          |        | RATT     |                                    | 2  |
| 3  |          | STOVL  | RONE     |                                    | 3  |
| 4  |          |        | VATT     |                                    | 4  |
| 5  |          | STCALL | VONE     | # GET SET FOR CONIC EXTRAP.,OTHER. | 5  |
| 6  |          |        | INTSTALL |                                    | 6  |
| 7  |          | SET    | DLOAD    |                                    | 7  |
| 8  |          |        | INTYPFLG |                                    | 8  |
| 9  |          |        | TAT      |                                    | 9  |
| 10 | OTHINT   | STORE  | TDEC1    |                                    | 10 |
| 11 |          | VLOAD  | CLEAR    |                                    | 11 |
| 12 |          |        | BASEOTP  |                                    | 12 |
| 13 |          |        | MOONFLAG |                                    | 13 |
| 14 |          | STOVL  | RCV      |                                    | 14 |
| 15 |          |        | BASEOTV  |                                    | 15 |
| 16 |          | STODL  | VCV      |                                    | 16 |
| 17 |          |        | BASETIME |                                    | 17 |
| 18 |          | BOF    | SET      |                                    | 18 |
| 19 |          |        | MOONTHIS |                                    | 19 |
| 20 |          |        | +2       |                                    | 20 |
| 21 |          |        | MOONFLAG |                                    | 21 |
| 22 |          | STCALL | TET      |                                    | 22 |
| 23 |          |        | INTEGRVS |                                    | 23 |
| 24 | COMPDISP | VLOAD  | VSU      |                                    | 24 |
| 25 |          |        | RATT     |                                    | 25 |
| 26 |          |        | RONE     |                                    | 26 |
| 27 |          | RTB    | PDDL     |                                    | 27 |
| 28 |          |        | NORMUNX1 | # UNIT(RANGE) TO PD 0-5            | 28 |
| 29 |          |        | 36D      |                                    | 29 |
| 30 |          | SL*    |          | # RESCALE AFTER NORMUNIT           | 30 |
| 31 |          |        | 0,1      |                                    | 31 |
| 32 |          | STOVL  | RANGE    | # SCALED 2(29)M                    | 32 |
| 33 |          |        | VATT     |                                    | 33 |
| 34 |          | VSU    | DOT      | # (VCM- VLM).UNIT(LOS). PD=0       | 34 |
| 35 |          |        | VONE     |                                    | 35 |
| 36 |          | SL1    |          | # SCALED 2(7)M/CS                  | 36 |
| 37 |          | STOVL  | RRATE    |                                    | 37 |
| 38 |          |        | RONE     |                                    | 38 |
| 39 |          | UNIT   | PDVL     | # UNIT(R) TO PD 0-5                | 39 |
| 40 |          |        | UNITZ    |                                    | 40 |
| 41 |          | CALL   |          |                                    | 41 |
| 42 |          |        | CDU*NBSM |                                    | 42 |
| 43 |          | VXM    | PUSH     | # UNIT (Z)/4 TO PD 6-11            | 43 |
| 44 |          |        | REFSMMAT |                                    | 44 |
| 45 |          | VPROJ  | VSL2     | # UNIT(P)=UNIT(UZ -(UZ)PROJ(UR))   | 45 |
| 46 |          |        | OD       |                                    | 46 |
| 47 |          | BVSU   | UNIT     |                                    | 47 |
| 48 |          |        | 6D       |                                    | 48 |
| 49 |          | PDVL   | VXV      | # UNIT(P) TO PD 12-17              | 49 |
| 50 |          |        | OD       | # UNIT(RL)                         | 50 |
| 51 |          |        | VONE     |                                    | 51 |
| 52 |          |        |          |                                    | 52 |
| 53 |          |        |          |                                    | 53 |
| 54 |          |        |          |                                    | 54 |
| 55 |          |        |          |                                    | 55 |
| 56 |          |        |          |                                    | 56 |
| 57 |          |        |          |                                    | 57 |
| 58 |          |        |          |                                    | 58 |
| 59 |          |        |          |                                    | 59 |
| 60 |          |        |          |                                    | 60 |

|    |        |        |           |                                           |
|----|--------|--------|-----------|-------------------------------------------|
| 1  |        |        |           |                                           |
| 2  |        | VXV    | DOT       | # (UR * VL)*UR . U(P)                     |
| 3  |        |        | 0D        |                                           |
| 4  |        |        | 12D       |                                           |
| 5  |        | PDVL   |           | # SIGN TO 12-13 , LOAD U(P)               |
| 6  |        | DOT    | SIGN      |                                           |
| 7  |        |        | 6D        |                                           |
| 8  |        |        | 12D       |                                           |
| 9  |        | SL2    | ACOS      | # ARCCOS(UP.UZ(SIGN))                     |
| 10 |        | STOVL  | RTHETA    |                                           |
| 11 |        |        | 0D        |                                           |
| 12 |        | DOT    | BPL       | # IF UR.UZ NEG,                           |
| 13 |        |        | 6D        | # RTHETA = 1 - RTHETA                     |
| 14 |        |        | +5        |                                           |
| 15 |        | DLOAD  | DSU       |                                           |
| 16 |        |        | DPPOS MAX |                                           |
| 17 |        |        | RTHETA    |                                           |
| 18 |        | STORE  | RTHETA    |                                           |
| 19 |        | EXIT   |           |                                           |
| 20 |        |        |           |                                           |
| 21 |        | CA     | BIT5      |                                           |
| 22 |        | MASK   | EXTVBACT  |                                           |
| 23 |        | EXTEND |           | # IF ANSWERED,                            |
| 24 |        | BZF    | ENDEXT    | # TERMINATE                               |
| 25 |        |        |           |                                           |
| 26 |        | CS     | EXTVBACT  |                                           |
| 27 |        | MASK   | BIT12     |                                           |
| 28 |        | ADS    | EXTVBACT  | # SET BIT 12                              |
| 29 |        | TCF    | REV83     | # AND START AGAIN.                        |
| 30 |        |        |           |                                           |
| 31 | GETRVN | CA     | PRI022    | # INHIBIT SERVICER                        |
| 32 |        | TC     | PRI0CHNG  |                                           |
| 33 |        | TC     | INTPRET   |                                           |
| 34 |        | VLOAD  | SETPD     |                                           |
| 35 |        |        | RN        | # LM STATE VECTOR IN RN,VN                |
| 36 |        |        | 0         |                                           |
| 37 |        | STOVL  | RONE      |                                           |
| 38 |        |        | VN        |                                           |
| 39 |        | STOVL  | VONE      | # LOAD R(CSM),V(CSM) IN CASE MUNFLAG SET  |
| 40 |        |        | V(CSM)    | # (TO INSURE TIME COMPATABILITY)          |
| 41 |        | PDVL   | PDDL      |                                           |
| 42 |        |        | R(CSM)    |                                           |
| 43 |        |        | PIPTIME   |                                           |
| 44 |        | EXIT   |           |                                           |
| 45 |        | CA     | PRI03     |                                           |
| 46 |        | TC     | PRI0CHNG  |                                           |
| 47 |        | TC     | INTPRET   |                                           |
| 48 |        | BOFF   | VLOAD     |                                           |
| 49 |        |        | MUNFLAG   |                                           |
| 50 |        |        | GETRVN2   | # IF MUNFLAG RESET, DO CM DELTA PRECISION |
| 51 |        |        |           |                                           |
| 52 |        |        |           |                                           |
| 53 |        |        |           |                                           |
| 54 |        |        |           |                                           |
| 55 |        |        |           |                                           |
| 56 |        |        |           |                                           |
| 57 |        |        |           |                                           |
| 58 |        |        |           |                                           |
| 59 |        |        |           |                                           |
| 60 |        |        |           |                                           |





# R31

PAGE 707

|    |         |        |          |                                          |    |
|----|---------|--------|----------|------------------------------------------|----|
| 1  |         |        |          |                                          | 1  |
| 2  |         | VXM    | VSR4     | # CHANGE TO REFERENCE SYSTEM AND RESCALE | 2  |
| 3  |         |        | REFSMMAT |                                          | 3  |
| 4  |         | PDVL   |          | # R TO PD 0-5                            | 4  |
| 5  |         | VXM    | VSL1     |                                          | 5  |
| 6  |         |        | REFSMMAT |                                          | 6  |
| 7  |         | PUSH   | SETPD    | # V TO PD 5-11                           | 7  |
| 8  |         |        | 0        |                                          | 8  |
| 9  |         | GOTO   |          |                                          | 9  |
| 10 |         |        | COMPDISP |                                          | 10 |
| 11 |         |        |          |                                          | 11 |
| 12 | GETRVN2 | CALL   |          |                                          | 12 |
| 13 |         |        | INTSTALL |                                          | 13 |
| 14 |         | CLEAR  | GOTO     |                                          | 14 |
| 15 |         |        | INTYPFLG | # PREC EXTRAP FOR OTHER                  | 15 |
| 16 |         |        | OTHINT   |                                          | 16 |
| 17 | R31SURF | TC     | INTPRET  |                                          | 17 |
| 18 |         | RTB    |          | # LM IS ON SURFACE, SO PRECISION         | 18 |
| 19 |         |        | LOADTIME | # INTEGRATION USES PLANETARY INERTIAL    | 19 |
| 20 |         | STCALL | TDEC1    | # ORIENTATION SUBROUTINE                 | 20 |
| 21 |         |        | LEMPREC  |                                          | 21 |
| 22 |         | GOTO   |          | # DO CSM CONIC                           | 22 |
| 23 |         |        | OTHCONIC |                                          | 23 |
| 24 | MUNG?   | CS     | FLAGWRD6 |                                          | 24 |
| 25 |         | MASK   | MUNFLBIT |                                          | 25 |
| 26 |         | EXTEND |          |                                          | 26 |
| 27 |         | BZF    | GETRVN   | # IF MUNFLAG SET, CSM BASE NOT NEEDED    | 27 |
| 28 |         |        |          |                                          | 28 |
| 29 | ONEBASE | TC     | INTPRET  | # GET CSM BASE VECTOR                    | 29 |
| 30 |         | RTB    | GOTO     |                                          | 30 |
| 31 |         |        | LOADTIME |                                          | 31 |
| 32 |         |        | DOCMBASE |                                          | 32 |
| 33 |         |        |          |                                          | 33 |
| 34 | V16N54  | VN     | 1654     |                                          | 34 |
| 35 |         |        |          |                                          | 35 |
| 36 |         |        |          |                                          | 36 |
| 37 |         |        |          |                                          | 37 |
| 38 |         |        |          |                                          | 38 |
| 39 |         |        |          |                                          | 39 |
| 40 |         |        |          |                                          | 40 |
| 41 |         |        |          |                                          | 41 |
| 42 |         |        |          |                                          | 42 |
| 43 |         |        |          |                                          | 43 |
| 44 |         |        |          |                                          | 44 |
| 45 |         |        |          |                                          | 45 |
| 46 |         |        |          |                                          | 46 |
| 47 |         |        |          |                                          | 47 |
| 48 |         |        |          |                                          | 48 |
| 49 |         |        |          |                                          | 49 |
| 50 |         |        |          |                                          | 50 |
| 51 |         |        |          |                                          | 51 |
| 52 |         |        |          |                                          | 52 |
| 53 |         |        |          |                                          | 53 |
| 54 |         |        |          |                                          | 54 |
| 55 |         |        |          |                                          | 55 |
| 56 |         |        |          |                                          | 56 |
| 57 |         |        |          |                                          | 57 |
| 58 |         |        |          |                                          | 58 |
| 59 |         |        |          |                                          | 59 |
| 60 |         |        |          |                                          | 60 |

1412THE



```
1) PROGRAM NAME - TARGET DELTA V PROGRAM (P76).
2) FUNCTIONAL DESCRIPTION - UPON ENTRY BY ASTRONAUT ACTION, P76 FLASHES DSKY REQUESTS TO THE ASTRONAUT
TO PROVIDE VIA DSKY (1) THE DELTA V TO BE APPLIED TO THE OTHER VEHICLE STATE VECTOR AND (2) THE
TIME (TIG) AT WHICH THE OTHER VEHICLE VELOCITY WAS CHANGED BY EXECUTION OF A THRUSTING MANEUVER. THE
OTHER VEHICLE STATE VECTOR IS INTEGRATED TO TIG AND UPDATED BY THE ADDITION OF DELTA V (DELTA V HAVING
BEEN TRANSFORMED FROM LV TO REF COSYS). USING INTEGRVS, THE PROGRAM THEN INTEGRATES THE OTHER
VEHICLE STATE VECTOR TO THE STATE VECTOR OF THIS VEHICLE, THUS INSURING THAT THE W-MATRIX AND BOTH VEHICLE
STATES CORRESPOND TO THE SAME TIME.
3) ERASABLE INITIALIZATION REQUIRED - NONE.
4) CALLING SEQUENCES AND EXIT MODES - CALLED BY ASTRONAUT REQUEST THRU DSKY V 37 E 76 E.
EXITS BY TCF ENDOFJOB.
5) OUTPUT -- OTHER VEHICLE STATE VECTOR INTEGRATED TO TIG AND INCREMENTED BY DELTA V IN REF COSYS.
THE PUSHLIST CONTAINS THE MATRIX BY WHICH THE INPUT DELTA V MUST BE POST-MULTIPLIED TO CONVERT FROM LV
TO REF COSYS.
6) DEBRIS - OTHER VEHICLE STATE VECTOR.
7) SUBROUTINES CALLED - BANKCALL, GOXDSPF, CSMPREC (OR LEMPREC), ATOPCSM (OR ATOPLEM), INTSTALL, INTWAKE, PHASCHNG
INTERPRET, INTEGRVS, AND MINIRECT.
8) FLAG USE - MOONFLAG, CMOONFLG, INTYPFLG, RASFLAG, AND MARKCTR.

 BANK 30
 SETLOC P76LOC
 BANK

 COUNT* $$/P76

 EBANK= TIG

P76 TC UPFLAG
 ADRES TRACKFLG

 TC INTERPRET
 VLOAD

 STORE DELVLVC
 EXIT DELVOV

 CAF V06N84 # FLASH LAST DELTA V,
 TC BANKCALL # AND WAIT FOR KEYBOARD ACTION.

 CADR GOFLASH
 TCF ENDP76
 TC +2 # PROCEED
 TC -5 # STORE DATA AND REPEAT FLASHING
 CAF V06N84 +1 # FLASH VERB 06 NOUN 33, DISPLAY LAST TIG,
 TC BANKCALL # AND WAIT FOR KEYBOARD ACTION.

 CADR GOFLASH
 TCF ENDP76
 TC +2
 TC -5
 TC INTERPRET # RETURN TO INTERPRETIVE CODE
```

|    |  |   |
|----|--|---|
| 75 |  |   |
| 76 |  | 1 |
| 77 |  |   |

[illegible]

```
1 # SUBROUTINE NAME: V82CALL
2
3 # MOD NO: 0 DATE: 16 FEB 67
4 # MOD BY: R. R. BAIRNSFATHER LOG SECTION: R30
5 # MOD NO: 1 MOD BY: R. R. BAIRNSFATHER DATE: 11 APR 67 SR30.1 CHANGED TO ALLOW MONITOR OPERN
6 # MOD NO: 2 MOD BY: ALONSO DATE: 11 DEC 67 VB82 PROGRAM REWRITTEN
7 # MOD NO: 3 MOD BY: ALONSO DATE: 26 MAR 68 PROG MOD TO HANDLE DIF EARTH/MOON SCALE
8
9 # NEW FUNCTIONAL DESCRIPTION: CALLED BY VERB 82 ENTER. PRIORITY 10.
10 # USED THROUGHOUT. CALCULATE AND DISPLAY ORBITAL PARAMETERS
11
12 # 1. IF AVERAGE G IS OFF:
13 # FLASH DISPLAY V04N06. R2 INDICATES WHICH SHIP'S STATE VECTOR IS
14 # TO BE UPDATED. INITIAL CHOICE IS THIS SHIP (R2=1). ASTRONAUT
15 # CAN CHANGE TO OTHER SHIP BY V22EXE, WHERE X NOT EQ 1.
16 # SELECTED STATE VECTOR UPDATED BY THISPREC (OTHPREC).
17 # CALLS SR30.1 (WHICH CALLS TFFCONMU + TFFRP/RA) TO CALCULATE
18 # RPER (PERIGEE RADIUS), RAPO (APOGEE RADIUS), HPER (PERIGEE
19 # HEIGHT ABOVE LAUNCH PAD OR LUNAR LANDING SITE), HARD (APOGEE
20 # HEIGHT AS ABOVE), TPER (TIME TO PERIGEE), TFF (TIME TO
21 # INTERSECT 300 KFT ABOVE PAD OR 35KFT ABOVE LANDING SITE).
22 # FLASH MONITOR V16N44 (HAPO, HPER, TFF).TFF IS -59M59S IF IT WAS
23 # NOT COMPUTABLE, OTHERWISE IT INCREMENTS ONCE PER SECOND.
24 # ASTRONAUT HAS OPTION TO MONITOR TPER BY KEYING IN N 32 E.
25 # DISPLAY IS IN HMS, IS NEGATIVE (AS WAS TFF), AND INCREMENTS
26 # ONCE PER SECOND ONLY IF TFF DISPLAY WAS -59M59S.
27
28 # 2. IF AVERAGE G IS ON:
29 # CALLS SR30.1 APPROX EVERY TWO SECS. STATE VECTOR IS ALWAYS
30 # FOR THIS VEHICLE. V82 DOES NOT DISTURB STATE VECTOR. RESULTS
31 # OF SR30.1 ARE RAPO, RPER, HAPO, HPER, TPER, TFF.
32 # FLASH MONITOR V16N44 (HAPO, HPER, TFF).
33 # ADDENDUM: HAPO AND HPER SHOULD BE CHANGED TO READ HAPOX AND HPERX IN THE
34 # ABOVE REMARKS.
35
36 # CALLING SEQUENCE: VERB 82 ENTER.
37
38 # SUBROUTINES CALLED: SR30.1, GOXDSPF
39 # MAYBE - THISPREC , OTHPREC, LOADTIME, DELRSPL
40 # NORMAL EXIT MODES: TC ENDEXT
41
42 # ALARMS: NONE
43
44 # OUTPUT: HAPOX (-29) M
45 # HPERX (-29) M
46 # RAPO (-29) M EARTH
47 # (-27) M MOON
48 # RPER (-29) M EARTH
49 # (-27) M MOON
50 # TFF (-28) CS CONTAINS NEGATIVE QUANTITY
51 # -TPER (-28) CS CONTAINS NEGATIVE QUANTITY
```

```
#
ERASABLE INITIALIZATION REQUIRED: STATE VECTOR.
#
DEBRIS: QPRET, RONE, VONE, TFF/RTMU, HPERMIN, RPADTEM, V82EMFLG.
MAYBE: TSTART82, V82FLAGS, TDEC1.

 EBANK= HAPOX
 BANK 31
 SETLOC R30LOC
 BANK
 COUNT* $$/R30

V82CALL TC INTERP
 BON GOTO
 AVEGFLAG
 V82GON # IF AVERAGE G ON
 V82GOFF # IF AVERAGE G OFF

V82GOFF EXIT
 CAF TWO # ALLOW ASTRONAUT TO SELECT VEHICLE
 TS OPTIONX # DESIRED FOR ORBITAL PARAMETERS
 CAF ONE # CALCULATION AND DISPLAY.
 TS OPTIONX +1
 CAF OPTIONVN # V 04 N 06
 TC BANKCALL
 CADR GOXDSPF
 TC ENDEXT # TERMINATE
 TC +2 # PROCEED
 TC -5 # DATA IN. OPTION1+1 = 1 FOR THIS VEHICLE.
 # UNEQ 1 FOR OTHER VEHICLE.
 CAF BIT4 # 80 MS
 TC WAITLIST
 EBANK= TFF
 2CADR TICKTEST

V82GOFLP RELINT
 CAF TFFBANK # MAJOR RECYCLE LOOP ENTRY
 TS EBANK
 CAF ZERO
 TS V82FLAGS # ZERO FLAGS FOR TICKTEST, INHIBITS
 # DECREMENTING OF TFF AND -TPER.
 CAF PRI07
 TC FINDVAC # V82GOFF1 WILL EXECUTE STATE VECTOR
 EBANK= TFF # UPDATE AND ORBIT CALCULATIONS FOR
 2CADR V82GOFF1 # SELECTED VEHICLE ABOUT PROPER BODY.

 RELINT
V82STALL CAF THREE # STALL IN THIS LOOP AND WITHOLD V 16 N 44
```

|    |          |        |            |                                            |
|----|----------|--------|------------|--------------------------------------------|
| 1  |          |        |            |                                            |
| 2  |          | MASK   | V82FLAGS   | # UNTIL STATE VECTOR UPDATE SETS ONE OF    |
| 3  |          | CCS    | A          | # OUR FLAG BITS.                           |
| 4  |          | TC     | FLAGGON    | # EXIT FROM STALL LOOP.                    |
| 5  |          | CAF    | 1SEC       |                                            |
| 6  |          | TC     | BANKCALL   |                                            |
| 7  |          | CADR   | DELAYJOB   |                                            |
| 8  |          | TC     | V82STALL   |                                            |
| 9  |          |        |            |                                            |
| 10 | FLAGGON  | CAF    | V16N44     | # MONITOR HAPO, HPER, TFF.                 |
| 11 |          | TC     | BANKCALL   |                                            |
| 12 |          | CADR   | GOXDSPF    |                                            |
| 13 |          | TC     | B5OFF      | # TERM THIS TELLS TICKTEST TO KILL ITSELF  |
| 14 |          | TC     | B5OFF      | # PROCEED DITTO                            |
| 15 |          | TC     | V82GOFLP   | # RECYCLE RECOMPUTE STATE VECT + DISPLAY   |
| 16 |          |        |            |                                            |
| 17 | OPTIONVN | VN     | 412        |                                            |
| 18 | V16N44   | VN     | 1644       |                                            |
| 19 | TFFBANK  | ECADR  | TFF        |                                            |
| 20 |          |        |            |                                            |
| 21 | V82GOFF1 | TC     | INTPRET    |                                            |
| 22 |          | RTB    |            |                                            |
| 23 |          |        | LOADTIME   |                                            |
| 24 |          | STORE  | TDEC1      | # TIME FOR STATE VECTOR UPDATE.            |
| 25 |          | STORE  | TSTART82   | # TIME FOR INTERNAL USE.                   |
| 26 |          | EXIT   |            |                                            |
| 27 |          | CS     | OPTIONX +1 | # 1 FOR THIS VEHICLE, NOT 1 FOR OTHER.     |
| 28 |          | AD     | ONE        |                                            |
| 29 |          | EXTEND |            |                                            |
| 30 |          | BZF    | THISSHIP   |                                            |
| 31 | OTHSHIP  | TC     | INTPRET    |                                            |
| 32 |          | CALL   |            | # CALL STATE VECTOR UPDATE FOR OTHER SHIP. |
| 33 |          |        | OTHPREC    |                                            |
| 34 | BOTHSHIP | VLOAD  |            | # MOVE RESULTS INTO TFFCONIC STORAGE AREAS |
| 35 |          |        | RATT       | # TO BE CALLED BY SR30.1.                  |
| 36 |          | STOVL  | RONE       | # RATT AT (-29)M FOR EARTH OR MOON         |
| 37 |          |        | VATT       |                                            |
| 38 |          | STORE  | VONE       | # VATT AT (-7)M/CS FOR EARTH OR MOON       |
| 39 |          | DLOAD* |            |                                            |
| 40 |          |        | 1/RTMUE,2  | # X2 IS 0 FOR EARTH CENTERED STATE VEC     |
| 41 |          | STORE  | TFF/RTMU   | # X2 IS 2 FOR MOON                         |
| 42 |          | DLOAD* |            | # AS LEFT BY THISPREC OR OTHPREC.          |
| 43 |          |        | MINPERE,2  |                                            |
| 44 |          | STORE  | HPERMIN    | # TFFRTMU, HPERMIN AND RPADTEM ARE ALL     |
| 45 |          | SLOAD  | BHIZ       | # EARTH/MOON PARAMETERS AS SET HERE.       |
| 46 |          |        | X2         |                                            |
| 47 |          |        | EARTHPAD   |                                            |
| 48 |          | GOTO   |            |                                            |
| 49 |          |        | MOONPAD    |                                            |
| 50 |          |        |            |                                            |
| 51 |          |        |            |                                            |
| 52 |          |        |            |                                            |
| 53 |          |        |            |                                            |
| 54 |          |        |            |                                            |
| 55 |          |        |            |                                            |
| 56 |          |        |            |                                            |
| 57 |          |        |            |                                            |
| 58 |          |        |            |                                            |
| 59 |          |        |            |                                            |
| 60 |          |        |            |                                            |



```
1 THISSHIP TC INTPRET
2 CALL # CALL STATE VECTOR UPDATE FOR THIS SHIP.
3
4 THISPREC
5 GOTO
6 BOTHSHIP
7
8 # THE FOLLOWING CONSTANTS ARE PAIRWISE INDEXED. DO NOT SEPARATE PAIRS.
9
10 MINPERM 2DEC 10668 B-27 # 35 KFT MIN PERIGEE HEIGHT FOR MOON(-27)M
11
12 MINPERE 2DEC 91440 B-29 # 300 KFT (-29)M FOR EARTH
13
14 EARTHPAD DLOAD CLRGO # PAD 37-B RADIUS. SCALED AT (-29)M.
15 RPAD
16 V82EMFLG # INDICATE EARTH SCALING FOR SR30.1
17 BOTHPAD
18
19 MOONPAD VLOAD ABVAL # COMPUTE MOON PAD RADIUS FROM RLS VECTOR.
20 RLS # SCALED AT (-27)M.
21 SET
22
23 BOTHPAD STCALL V82EMFLG # INDICATE MOON SCALING FOR SR30.1
24 RPADTEM # CALCULATE ORBITAL PARAMETERS
25 SR30.1
26 RTB DSU
27 LOADTIME
28 TSTART82 # PRESENT TIME - TIME V82GOFF1 BEGAN
29
30 STORE # SAVE IT
31 DLOAD BZE
32 # SR30.1 SETS -TPER=0 IF HPER L/
33 # HPERMIN (300 OR 35) KFT.
34 TICKTPER DLOAD TICKTIFF # (-TPER = 0)
35 DAD # (-TPER NON ZERO) TFF WAS NOT COMPUTED.
36 -TPER # BUT WAS SET TO 59M59S.DONT TICK TFF, DO
37 TSTART82 # TICK -TPER. DISPLAY BOTH.
38 -TPER # -TPER CORRECTED FOR TIME SINCE V82GOFF1
39 # BEGAN.
40
41 CAF BIT1
42 TS V82FLAGS
43 TC ENDOFJOB # INFORMS TICKTEST TO INCREMENT ONLY -TPER
44
45 TICKTIFF DLOAD DAD # (-TPER=0) TFF WAS COMPUTED.TICK TFF.
46 TFF # DO NOT TICK -TPER.DISPLAY TFF, BUT NOT
47 TSTART82 # -TPER.
48 TFF # TFF CORRECTED FOR TIME SINCE V82GOFF1
49 # BEGAN.
50
51 CAF BIT2
52 TS V82FLAGS
53 TC ENDOFJOB # INFORMS TICKTEST TO INCREMENT ONLY TFF.
```

# R30

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  | TICKTEST | CAF    | BIT5     | # THIS WAITLIST PROGRAM PERPETUATES ITSELF | 2  |
| 3  |          | MASK   | EXTVBACT | # ONCE A SEC UNTIL BIT 5 OF EXTVBACT =0.   | 3  |
| 4  |          | CCS    | A        |                                            | 4  |
| 5  |          | TC     | DOTICK   |                                            | 5  |
| 6  |          | CAF    | PRI025   |                                            | 6  |
| 7  |          | TC     | NOVAC    | # TERMINATE V 82.CANT CALL ENDEXT IN RUPT. | 7  |
| 8  |          | EBANK= | EXTVBACT |                                            | 8  |
| 9  |          | 2CADR  | ENDEXT   |                                            | 9  |
| 10 |          |        |          |                                            | 10 |
| 11 |          | TC     | TASKOVER |                                            | 11 |
| 12 | DOTICK   | CAF    | 1SEC     | # RE-REQUEST TICKTEST.                     | 12 |
| 13 |          | TC     | WAITLIST |                                            | 13 |
| 14 |          | EBANK= | TFF      |                                            | 14 |
| 15 |          | 2CADR  | TICKTEST |                                            | 15 |
| 16 |          |        |          |                                            | 16 |
| 17 |          | CAF    | THREE    |                                            | 17 |
| 18 |          | MASK   | V82FLAGS |                                            | 18 |
| 19 |          | INDEX  | A        |                                            | 19 |
| 20 |          | TC     | +1       |                                            | 20 |
| 21 |          | TC     | TASKOVER | # IF NO FLAGBITS SET DONT CHANGE TFF OR    | 21 |
| 22 |          |        |          | # -TPER, BUT CONTINUE LOOP.                | 22 |
| 23 |          | TC     | TPERTICK | # ONLY BIT 1 SET. INCR -TPER BY 1 SEC.     | 23 |
| 24 | TFFTICK  | CAF    | 1SEC     | # ONLY BIT 2 SET. INCR TFF BY 1 SEC.       | 24 |
| 25 |          | TS     | L        |                                            | 25 |
| 26 |          | CAF    | ZERO     |                                            | 26 |
| 27 |          | DAS    | TFF      |                                            | 27 |
| 28 |          | TC     | TASKOVER |                                            | 28 |
| 29 | TPERTICK | CAF    | 1SEC     |                                            | 29 |
| 30 |          | TS     | L        |                                            | 30 |
| 31 |          | CAF    | ZERO     |                                            | 31 |
| 32 |          | DAS    | -TPER    |                                            | 32 |
| 33 |          | TC     | TASKOVER |                                            | 33 |
| 34 |          |        |          |                                            | 34 |
| 35 |          |        |          |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 |          |        |          |                                            | 37 |
| 38 |          |        |          |                                            | 38 |
| 39 |          |        |          |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 |          |        |          |                                            | 41 |
| 42 |          |        |          |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          |        |          |                                            | 44 |
| 45 |          |        |          |                                            | 45 |
| 46 |          |        |          |                                            | 46 |
| 47 |          |        |          |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          |        |          |                                            | 49 |
| 50 |          |        |          |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

V82GON

EXIT

# AVERAGE G ON. USE CURRENT STATE VECTOR  
# FOR ORBITAL PARAMETER CALCULATIONS.CAF PRI07  
TC FINDVAC  
EBANK= TFF# LESS THAN LAMBERT  
# V82GON1 WILL PERFORM ORBIT CALCULATIONS  
# ABOUT PROPER BODY APPROX ONCE PER SEC.

2CADR V82GON1

RELINT

CCS NEWJOB  
TC CHANG1# WITHOLD V16 N44 UNTIL FIRST ORBIT CALC  
# IS DONE. NOTE: V82GON1 (PRI07, FINDVAC  
# JOB) IS COMPLETED BEFORE V82GON (PRI07,  
# NOVAC JOB).

V82REDSP

CAF V16N44  
TC BANKCALL

# MONITOR HAPO, HPER, TFF

CADR GOXDSPF  
TC B5OFF  
TC B5OFF# TERM THIS TELLS V82GON1 TO KILL ITSELF.  
# PROC DITTO.

TC V82REDSP

# RECYCLE

V82GON1

TC

INTPRET

# THIS EXEC PROGRAM PERPETUATES ITSELF  
# ONCE A SEC UNTIL BIT 5 OF EXTVBACT =0.VLOAD GOTO  
RN# HOLDS OFF CCS NEWJOB BETWEEN RN AND  
# VN FETCH SO RN , VN ARE FROM SAME

NEXTLINE

STOVL

NEXTLINE  
RONE  
VN# STATE VECTOR UPDATE.  
# RN AT (-29)M FOR EARTH OR MOONSTORE VONE  
BON GOTO

# VN AT (-7)M/CS FOR EARTH OR MOON

MOONTHIS  
MOONGON  
EARTHGON# FLAG INDICATES BODY ABOUT WHICH ORBITAL  
# CALCULATIONS ARE TO BE PERFORMED.  
# IF SET - MOON , IF RESET - EARTH.

MOONGON

SET

DLOAD  
V82EMFLG  
1/RTMUM# INDICATE MOON SCALING FOR SR30.1  
# LUNAR PARAMETERS LOADED HERE FOR SR30.1STODL TFF/RTMU  
MINPERM

STOVL HPERMIN

ABVAL RLS  
GOTO  
V82GON2

# SCALED AT (-27)M.

EARTHGON

CLEAR

DLOAD  
V82EMFLG  
1/RTMUE# INDICATE EARTH SCALING FOR SR30.1  
# EARTH PARAMETERS LOADED HERE FOR SR30.1STODL TFF/RTMU  
MINPERE

STODL HPERMIN

V82GON2

STCALL

RPAD  
RPADTEM  
SR30.1

# COMMON CODE FOR EARTH &amp; MOON.

|         |                |          |                                            |
|---------|----------------|----------|--------------------------------------------|
| V82GON3 | EXIT<br>CAF    | BIT5     |                                            |
|         | MASK<br>EXTEND | EXTVBACT | # SEE IF ASTRONAUT HAS SIGNALLED TERMINATE |
|         | BZF            | ENDEXT   | # YES, TERMINATE VB 82 LOOP                |
|         | CAF            | 1SEC     |                                            |
|         | TC             | BANKCALL | # WAIT ONE SECOND BEFORE REPEATING         |
|         | CADR           | DELAYJOB | # ORBITAL PARAMETER COMPUTATION.           |
|         | TC             | V82GON1  |                                            |
| SPLRET  | =              | V82GON3  |                                            |

```
1 # SUBROUTINE NAME: SR30.1
2
3 # MOD NO: 0 DATE: 16 FEB 67
4 # MOD BY: R. R. BAIRNSFATHER LOG SECTION: R32
5 # MOD NO: 1 MOD BY: R. R. BAIRNSFATHER DATE: 11 APR 67 SR30.1 CHANGED TO ALLOW MONITOR OPERN
6 # MOD NO: 2 MOD BY: R. R. BAIRNSFATHER DATE: 14 APR 67 ADD OVFL CK FOR RAPO
7 # MOD NO: 3 MOD BY ALONSO DATE: 11 DEC 67 SUBROUTINE REWRITTEN
8 # MOD NO: 4 MOD BY ALONSO DATE: 26 MAR 68 PROG MOD TO HANDLE DIF EARTH/MOON SCALE
9 # MOD NO: 5 MOD BY: R. R. BAIRNSFATHER DATE: 6 AUG 68 OVFL CK FOR HAPO & HPER. VOIDS MOD #2.
10
11 #
12 # NEW FUNCTIONAL DESCRIPTION: ORBITAL PARAMETERS DISPLAY FOR NOUNS 32 AND 44.
13 # SR30.1 CALLS TFFCONMU AND TFFRP/RA TO CALCULATE RPER (PERIGEE RADIUS),
14 # RAPO (APOGEE RADIUS), HPER (PERIGEE HEIGHT ABOVE LAUNCH PAD OR LUNAR
15 # LANDING SITE), HAPO (APOGEE HEIGHT AS ABOVE), TPER (TIME TO PERIGEE),
16 # TFF (TIME TO INTERSECT 300 KFT ABOVE PAD OR 35KFT ABOVE LANDING SITE).
17 # IF HPER IS GREATER THAN OR EQUAL TO HPERMIN, CALCULATES TPER AND STORES
18 # NEGATIVE IN -TPER. OTHERWISE STORES +0 IN -TPER. WHENEVER TPER IS
19 # CALCULATED, TFF IS NOT COMPUTABLE AND DEFAULTS TO -59MIN 59SEC. IF HAPO
20 # WOULD EXCEED 9999.9 NM, IT IS LIMITED TO THAT VALUE FOR DISPLAY.
21 #
22 # ADDENDUM: HAPO AND HPER SHOULD BE CHANGED TO READ HAPOX AND HPERX IN THE
23 # ABOVE REMARKS.
24 #
25 # CALLING SEQUENCE: CALL
26 # SR30.1
27 # SUBROUTINES CALLED: TFFCONMU, TFFRP/RA, CALCTPER, CALCTFF
28 #
29 # NORMAL EXIT MODE: CALLING LINE +1 (STILL IN INTERPRETIVE MODE)
30 #
31 # ALARMS: NONE
32 #
33 # OUTPUT: RAPO (-29) M EARTH APOGEE RADIUS EARTH CENTERED COORD.
34 # (-27) M MOON MOON CENTERED COORD.
35 # RPER (-29) M EARTH PERIGEE RADIUS EARTH CENTERED COORD.
36 # (-27) M MOON MOON CENTERED COORD.
37 # HAPOX (-29) M APOGEE ALTITUDE ABOVE PAD OR LAND. SITE MAX VALUE LIMITED TO 9999.9 NM.
38 # HPERX (-29) M PERIGEE ALT. ABOVE PAD OR LAND. SITE MAX VALUE LIMITED TO 9999.9 NM.
39 # TFF (-28) CS TIME TO 300KFT OR 35KFT ALTITUDE
40 # -TPER (-28) CS TIME TO PERIGEE
41 #
42 # ERASABLE INITIALIZATION REQUIRED --
43 # TFF/RTMU (+17) EARTH RECIPROCAL OF PROPER GRAV CONSTANT FOR
44 # (+14) MOON EARTH OR MOON = 1/SQRT(MU).
45 # RONE (-29) M STATE VECTOR
46 # VONE (-7) M/CS STATE VECTOR
47 # RPADTEM (-29) M EARTH RADIUS OF LAUNCH PAD OR LUNAR LANDING
48 # (-27) M MOON SITE.
49 # HPERMIN (-29) M EARTH (300 OR 35) KFT MINIMUM PERIGEE ALTITUDE
50 # (-27) M MOON ABOVE LAUNCH PAD OR LUNAR LANDING SITE.
51 # V82EMFLG (INT SW BIT) RESET FOR EARTH, SET FOR MOON.
52 #
53 # DEBRIS: QPREG, PDL, S2
```

```
1
2 COUNT* $$/SR30S
3
4 SR30.1 SETPD STQ # INITIALIZE PUSHDOWN LIST.
5 0
6 S2
7
8 # SR30.1 INPUT: RONE AT (-29)M EARTH/MOON
9 # VONE AT (-7)M/CS
10 # TFFCONMU, TFFRP/RA, CALCTPER, AND CALCTFF
11 # CALLS REQUIRE:
12 # EARTH CENTERED (NO RESCALING REQUIRED)
13 # RONE SCALED TO B-29 M
14 # VONE SCALED TO B-7 M/CS
15 # MOON CENTERED (RESCALING REQUIRED)
16 # RONE SCALED TO B-27 M
17 # VONE SCALED TO B-5 M/CS
18
19 BOFF VLOAD
20 V82EMFLG # OFF FOR EARTH, ON FOR MOON.
21 TFFCALLS
22 RONE
23
24 VSL2
25 STOVL RONE
26 VONE
27
28 VSL2
29 STORE VONE
30 CALL
31 TFFCONMU
32
33 CALL # TFFRP/RA COMPUTES RAPO,RPER.
34 TFFRP/RA
35 # RETURNS WITH RAPO IN D(MPAC).
36
37 DSU
38
39 BOFF RPADTEM
40 SR2R # NEED HAPO AT (-29)M FOR DISPLAY.
41
42 # IF MOON CENTERED, RESCALE FROM (-27)M.
43 # IF EARTH CENTERED ALREADY AT (-29)M.
44 V82EMFLG
45 +1 # OFF FOR EARTH, ON FOR MOON.
46
47 CALL # IF RAPO > MAXNM, SET RAPO =9999.9 NM.
48 MAXCHK # OTHERWISE STORE (RAPO-RPADTEM) IN HAPO.
49
50 STORHAPO STODL HAPOX
51 RPER
52
53 DSU
54
55 RPADTEM # GIVES HPER AT (-29)M EARTH, (-27)M MOON.
56 MPAC +4 # SAVE THIS FOR COMPARISON TO HPERMIN.
57 SR2R # NEED HPER AT (-29)M FOR DISPLAY.
58
59 # IF MOON CENTERED, RESCALE FROM (-27)M.
60 # IF EARTH CENTERED ALREADY AT (-29)M.
61 V82EMFLG
62 +1 # OFF FOR EARTH, ON FOR MOON.
63
64 CALL # IF HPER > MAXNM, SET HPER = 9999.9 NM.
65 MAXCHK
```

|    |          |        |            |    |                                             |    |
|----|----------|--------|------------|----|---------------------------------------------|----|
| 1  |          |        |            |    |                                             | 1  |
| 2  | STORHPER | STODL  | HPERX      |    | # STORE (RPER - RPADEM) INTO HPERX.         | 2  |
| 3  |          |        | MPAC       | +4 |                                             | 3  |
| 4  |          | DSU    | BPL        |    | # HPERMIN AT (-29)M FOR EARTH, (-27)M MOON  | 4  |
| 5  |          |        | HPERMIN    |    | # IF HPER L/ HPERMIN (300 OR 35)KFT,        | 5  |
| 6  |          |        | DOTPER     |    | # THEN ZERO INTO -TPER.                     | 6  |
| 7  |          | DLOAD  | GOTO       |    | # OTHERWISE CALCULATE TPER.                 | 7  |
| 8  |          |        | HI6ZEROS   |    |                                             | 8  |
| 9  |          |        | SKIPTPER   |    |                                             | 9  |
| 10 | DOTPER   | DLOAD  | CALL       |    |                                             | 10 |
| 11 |          |        | RPER       |    |                                             | 11 |
| 12 |          |        | CALCTPER   |    |                                             | 12 |
| 13 |          | DCOMP  |            |    | # TPER IS PUT NEG INTO -TPER.               | 13 |
| 14 | SKIPTPER | STODL  | -TPER      |    |                                             | 14 |
| 15 |          |        | HPERMIN    |    | # HPERMIN AT (-29)M FOR EARTH, (-27)M MOON  | 15 |
| 16 |          | DAD    | CALL       |    |                                             | 16 |
| 17 |          |        | RPADEM     |    | # RPADEM AT (-29)M FOR EARTH, (-27)M MOON   | 17 |
| 18 |          |        | CALCTFF    |    | # GIVES 59M59S FOR TFF IF HPER G/           | 18 |
| 19 |          | DCOMP  |            |    | # HPERMIN + RPADEM. (TPER WAS NON ZERO)     | 19 |
| 20 |          | STCALL | TFF        |    | # OTHERWISE COMPUTES TFF. (GOTO)            | 20 |
| 21 |          |        | S2         |    |                                             | 21 |
| 22 |          |        |            |    |                                             | 22 |
| 23 | MAXCHK   | DSU    | BPL        |    | # IF C(MPAC) > 9999.9 NM. MPAC = 9999.9 NM. | 23 |
| 24 |          |        | MAXNM      |    |                                             | 24 |
| 25 |          |        | +3         |    | # OTHERWISE C(MPAC) = B(MPAC).              | 25 |
| 26 |          | DAD    | RVQ        |    |                                             | 26 |
| 27 |          |        | MAXNM      |    |                                             | 27 |
| 28 | +3       | DLOAD  | RVQ        |    | # (USED BY P30 - P37 ALSO)                  | 28 |
| 29 |          |        | MAXNM      |    |                                             | 29 |
| 30 |          |        |            |    |                                             | 30 |
| 31 | MAXNM    | 2OCT   | 0106505603 |    |                                             | 31 |
| 32 |          |        |            |    |                                             | 32 |
| 33 |          |        |            |    |                                             | 33 |
| 34 |          |        |            |    |                                             | 34 |
| 35 |          |        |            |    |                                             | 35 |
| 36 |          |        |            |    |                                             | 36 |
| 37 |          |        |            |    |                                             | 37 |
| 38 |          |        |            |    |                                             | 38 |
| 39 |          |        |            |    |                                             | 39 |
| 40 |          |        |            |    |                                             | 40 |
| 41 |          |        |            |    |                                             | 41 |
| 42 |          |        |            |    |                                             | 42 |
| 43 |          |        |            |    |                                             | 43 |
| 44 |          |        |            |    |                                             | 44 |
| 45 |          |        |            |    |                                             | 45 |
| 46 |          |        |            |    |                                             | 46 |
| 47 |          |        |            |    |                                             | 47 |
| 48 |          |        |            |    |                                             | 48 |
| 49 |          |        |            |    |                                             | 49 |
| 50 |          |        |            |    |                                             | 50 |
| 51 |          |        |            |    |                                             | 51 |
| 52 |          |        |            |    |                                             | 52 |
| 53 |          |        |            |    |                                             | 53 |
| 54 |          |        |            |    |                                             | 54 |
| 55 |          |        |            |    |                                             | 55 |
| 56 |          |        |            |    |                                             | 56 |
| 57 |          |        |            |    |                                             | 57 |
| 58 |          |        |            |    |                                             | 58 |
| 59 |          |        |            |    |                                             | 59 |
| 60 |          |        |            |    |                                             | 60 |





## # STABLE ORBIT RENDEZVOUS PROGRAMS (P38 AND P78)

# MOD NO -1 LOG SECTION - STABLE ORBIT - P38-P39  
# MOD BY RUDNICKI.S DATE 25JAN68

## # FUNCTIONAL DESCRIPTION

# P38 AND P78 CALCULATE THE REQUIRED DELTA V AND OTHER INITIAL  
# CONDITIONS REQUIRED BY THE AGC TO (1) PUT THE ACTIVE VEHICLE  
# ON A TRANSFER TRAJECTORY THAT INTERCEPTS THE PASSIVE VEHICLE  
# ORBIT A GIVEN DISTANCE, DELTA R, EITHER AHEAD OF OR BEHIND THE  
# PASSIVE VEHICLE AND (2) ACTUALLY PLACE THE ACTIVE VEHICLE IN THE  
# PASSIVE VEHICLE ORBIT WITH A DELTA R SEPARATION BETWEEN THE TWO  
# VEHICLES

## # CALLING SEQUENCE

# ASTRONAUT REQUEST THRU DSKY

# V37E38E IF THIS VEHICLE IS ACTIVE VEHICLE  
# V37E78E IF OTHER VEHICLE IS ACTIVE VEHICLE

## # INPUT

## # (1) SOI MANEUVER

# (A) TIG TIME OF SOI MANEUVER  
# (B) CENTANG ORBITAL CENTRAL ANGLE OF THE PASSIVE VEHICLE  
# DURING THE TRANSFER FROM TIG TO TIME OF INTERCEPT  
# (C) DELTAR THE DESIRED SEPARATION OF THE TWO VEHICLES  
# SPECIFIED AS A DISTANCE ALONG THE PASSIVE VEHICLE  
# ORBIT  
# (D) OPTION EQUALS 1 FOR SOI

## # (2) SOR MANEUVER

# (A) TIG TIME OF SOR MANEUVER  
# (B) CENTANG AN OPTIONAL RESPECIFICATION OF 1 (B) ABOVE  
# (C) OPTION EQUALS 2 FOR SOR  
# (D) DELTTIME THE TIME REQUIRED TO TRAVERSE DELTA R WHEN  
# TRAVELING AT A VELOCITY EQUAL TO THE HORIZONTAL  
# VELOCITY OF THE PASSIVE VEHICLE - SAVED FROM  
# SOI PHASE  
# (E) TINT TIME OF INTERCEPT (SOI) - SAVED FROM SOI PHASE

## # OUTPUT

# (1) TRKMKCNT NUMBER OF MARKS  
# (2) TTOGO TIME TO GO  
# (3) +MGA MIDDLE GIMBAL ANGLE

# STABLE\_ORBIT

|   |     |         |                                                                          |
|---|-----|---------|--------------------------------------------------------------------------|
| # | (4) | DSPTM1  | TIME OF INTERCEPT OF PASSIVE VEHICLE ORBIT<br>(FOR SOI ONLY)             |
| # | (5) | POSTTPI | PERIGEE ALTITUDE OF ACTIVE VEHICLE ORBIT AFTER<br>THE SOI (SOR) MANEUVER |
| # | (6) | DELVTPI | MAGNITUDE OF DELTA V AT SOI (SOR) TIME                                   |
| # | (7) | DELVTPF | MAGNITUDE OF DELTA V AT INTERCEPT TIME                                   |
| # | (8) | DELVLVC | DELTA VELOCITY AT SOI (AND SOR) - LOCAL VERTICAL<br>COORDINATES          |

# SUBROUTINE USED

|   |          |
|---|----------|
| # | AVFLAGA  |
| # | AVFLAGP  |
| # | VNDSPLY  |
| # | BANKCALL |
| # | GOFLASHR |
| # | GOTOPOOH |
| # | BLANKET  |
| # | ENDOFJOB |
| # | PREC/TT  |
| # | SELECTMU |
| # | INTRPVP  |
| # | MAINRTNE |

|        |          |
|--------|----------|
| BANK   | 04       |
| SETLOC | STBLEORB |
| BANK   |          |

|        |            |
|--------|------------|
| EBANK= | SUBEXIT    |
| COUNT* | \$\$/P3879 |

|     |      |          |                                   |
|-----|------|----------|-----------------------------------|
| P38 | TC   | BANKCALL |                                   |
|     | CADR | AVFLAGA  | # THIS VEHICLE ACTIVE             |
|     | TC   | +3       |                                   |
| P78 | TC   | BANKCALL |                                   |
|     | CADR | AVFLAGP  | # OTHER VEHICLE ACTIVE            |
|     | TC   | BANKCALL |                                   |
|     | CADR | P20FLGON | # SET UPDATFLG, TRACKFLG          |
|     | CAF  | DECTWO   |                                   |
|     | TS   | NN       |                                   |
|     | CAF  | V06N33SR | # DISPLAY TIG                     |
|     | TC   | VNDSPLY  |                                   |
|     | CAF  | V06N55SR | # DISPLAY CENTANG                 |
|     | TCR  | BANKCALL |                                   |
|     | CADR | GOFLASHR |                                   |
|     | TCF  | GOTOPOOH | # TERMINATE                       |
|     | TCF  | +5       | # PROCEED                         |
|     | TCF  | -5       | # RECYCLE                         |
|     | CAF  | THREE    | # IMMEDIATE RETURN - BLANK R1, R2 |
|     | TCR  | BLANKET  |                                   |

|    |         |        |          |                                          |
|----|---------|--------|----------|------------------------------------------|
| 1  |         |        |          |                                          |
| 2  |         | TCF    | ENDOFJOB |                                          |
| 3  |         | CAF    | FIVE     |                                          |
| 4  |         | TS     | OPTION1  |                                          |
| 5  |         | CAF    | ONE      |                                          |
| 6  |         | TS     | OPTION2  | # OPTION CODE IS SET TO 1                |
| 7  |         | CAF    | V04N06SR | # DISPLAY OPTION CODE - 1 = SOI, 2 = SOR |
| 8  |         | TCR    | BANKCALL |                                          |
| 9  |         | CADR   | GOFLASHR |                                          |
| 10 |         | TCF    | GOTOP00H | # TERMINATE                              |
| 11 |         | TCF    | +5       | # PROCEED                                |
| 12 |         | TCF    | -5       | # RECYCLE                                |
| 13 |         | CAF    | BIT3     | # IMMEDIATE RETURN - BLANK R3            |
| 14 |         | TCR    | BLANKET  |                                          |
| 15 |         | TCF    | ENDOFJOB |                                          |
| 16 |         | TC     | INTPRET  |                                          |
| 17 |         | SLOAD  | SR1      |                                          |
| 18 |         |        | OPTION2  |                                          |
| 19 |         | BHIZ   | DLOAD    |                                          |
| 20 |         |        | OPTN1    |                                          |
| 21 |         |        | TINT     |                                          |
| 22 |         | STORE  | TINTSOI  | # STORE FOR SOR PHASE                    |
| 23 |         | CLRGO  |          |                                          |
| 24 |         |        | OPTNSW   | # OPTNSW: ON = SOI, OFF = SOR            |
| 25 | OPTN1   | SET    | JUNCTN1  |                                          |
| 26 |         |        | CLEAR    | # SOI                                    |
| 27 |         |        | OPTNSW   |                                          |
| 28 |         |        | UPDATFLG |                                          |
| 29 |         | CALL   |          |                                          |
| 30 |         |        | PREC/TT  |                                          |
| 31 |         | DAD    | SET      |                                          |
| 32 |         |        | TIG      |                                          |
| 33 |         |        | UPDATFLG |                                          |
| 34 |         | STORE  | TINT     | # TI = TIG + TF                          |
| 35 |         | EXIT   |          |                                          |
| 36 |         | CAF    | V06N57SR | # DISPLAY DELTA R                        |
| 37 |         | TCR    | BANKCALL |                                          |
| 38 |         | CADR   | GOFLASHR |                                          |
| 39 |         | TCF    | GOTOP00H | # TERMINATE                              |
| 40 |         | TCF    | +5       | # PROCEED                                |
| 41 |         | TCF    | -5       | # RECYCLE                                |
| 42 |         | CAF    | SIX      | # IMMEDIATE RETURN - BLANK R2, R3        |
| 43 |         | TCR    | BLANKET  |                                          |
| 44 |         | TCF    | ENDOFJOB |                                          |
| 45 | +5      | EXTEND |          |                                          |
| 46 |         | DCA    | TINT     |                                          |
| 47 |         | DXCH   | DSPTM1   | # FOR DISPLAY                            |
| 48 |         | CAF    | V06N34SR | # DISPLAY TIME OF INTERCEPT              |
| 49 |         | TC     | VNDSPLY  |                                          |
| 50 |         | TC     | INTPRET  |                                          |
| 51 | JUNCTN1 | CLEAR  | CALL     |                                          |

|    |                      |        |           |                                           |    |
|----|----------------------|--------|-----------|-------------------------------------------|----|
| 1  |                      |        |           |                                           | 1  |
| 2  |                      |        | P39/79SW  |                                           | 2  |
| 3  |                      |        | SELECTMU  | # SELECT MU, CLEAR FINALFLG, GO TO VN1645 | 3  |
| 4  | RECYCLE              | CALL   |           |                                           | 4  |
| 5  |                      |        | PREC/TT   |                                           | 5  |
| 6  |                      | BOFF   | DLOAD     |                                           | 6  |
| 7  |                      |        | OPTNSW    |                                           | 7  |
| 8  |                      |        | OPTN2     |                                           | 8  |
| 9  |                      |        | TINT      |                                           | 9  |
| 10 |                      | STCALL | TDEC1     | # PRECISION UPDATE PASSIVE VEHICLE TO     | 10 |
| 11 |                      |        | INTRPVP   | # INTERCEPT TIME                          | 11 |
| 12 |                      | VLOAD  | UNIT      |                                           | 12 |
| 13 |                      |        | RATT      | # RP/(RP)                                 | 13 |
| 14 |                      | PDVL   | VXV       |                                           | 14 |
| 15 |                      |        | VATT      |                                           | 15 |
| 16 |                      | ABVAL  | NORM      | # (VP X RP/(RP))                          | 16 |
| 17 |                      |        | X1        |                                           | 17 |
| 18 |                      | PDDL   | DDV       |                                           | 18 |
| 19 |                      |        | DELTAR    |                                           | 19 |
| 20 |                      | SL*    |           | # DELTA R / (VP X RP/RP)                  | 20 |
| 21 |                      |        | 0 -7,1    |                                           | 21 |
| 22 |                      | STCALL | DELTIME   | # DELTA T = (RP) DELTA R / (VP X RP)      | 22 |
| 23 |                      |        | JUNCTN2   |                                           | 23 |
| 24 | OPTN2                | DLOAD  | DAD       |                                           | 24 |
| 25 |                      |        | TINTSOI   |                                           | 25 |
| 26 |                      |        | T         |                                           | 26 |
| 27 |                      | STORE  | TINT      | # TI = TI + TF                            | 27 |
| 28 | JUNCTN2              | DLOAD  | DSU       |                                           | 28 |
| 29 |                      |        | TINT      |                                           | 29 |
| 30 |                      |        | DELTIME   |                                           | 30 |
| 31 |                      | STORE  | TARGETIME | # TT = TI - DELTA T                       | 31 |
| 32 |                      |        |           |                                           | 32 |
| 33 | # .... MAINRTNE .... |        |           |                                           | 33 |
| 34 | # SUBROUTINES USED:  |        |           |                                           | 34 |
| 35 | #                    |        |           |                                           | 35 |
| 36 | # S3435.25           |        |           |                                           | 36 |
| 37 | # PERIAP01           |        |           |                                           | 37 |
| 38 | # SHIFTR1            |        |           |                                           | 38 |
| 39 | # VNDSPLY            |        |           |                                           | 39 |
| 40 | # BANKCALL           |        |           |                                           | 40 |
| 41 | # GOFLASH            |        |           |                                           | 41 |
| 42 | # GOTOP00H           |        |           |                                           | 42 |
| 43 | # VN1645             |        |           |                                           | 43 |
| 44 |                      |        |           |                                           | 44 |
| 45 | MAINRTNE             | STCALL | TDEC1     | # PRECISION UPDATE PASSIVE VEHICLE TO     | 45 |
| 46 |                      |        | INTRPVP   | # TARGET TIME                             | 46 |
| 47 |                      | DLOAD  |           |                                           | 47 |
| 48 |                      |        | TIG       |                                           | 48 |
| 49 |                      | STORE  | INTIME    |                                           | 49 |
| 50 |                      | SSP    | VLOAD     |                                           | 50 |
| 51 |                      |        | SUBEXIT   |                                           | 51 |
| 52 |                      |        | TEST3979  |                                           | 52 |
| 53 |                      |        |           |                                           | 53 |
| 54 |                      |        |           |                                           | 54 |
| 55 |                      |        |           |                                           | 55 |
| 56 |                      |        |           |                                           | 56 |
| 57 |                      |        |           |                                           | 57 |
| 58 |                      |        |           |                                           | 58 |
| 59 |                      |        |           |                                           | 59 |
| 60 |                      |        |           |                                           | 60 |

1412THE

|    |                                                |                                      |                                        |                        |    |
|----|------------------------------------------------|--------------------------------------|----------------------------------------|------------------------|----|
| 1  |                                                |                                      |                                        | 1                      |    |
| 2  | RATT                                           |                                      |                                        | 2                      |    |
| 3  | CALL                                           |                                      |                                        | 3                      |    |
| 4  | TEST3979                                       | BOFF                                 | S3435.25                               | 4                      |    |
| 5  |                                                |                                      | BON                                    | 5                      |    |
| 6  |                                                |                                      | P39/79SW                               | 6                      |    |
| 7  |                                                |                                      |                                        | 7                      |    |
| 8  |                                                |                                      |                                        | 8                      |    |
| 9  |                                                |                                      |                                        | 9                      |    |
| 10 |                                                |                                      |                                        | 10                     |    |
| 11 | SET                                            |                                      |                                        |                        | 11 |
| 12 | P39P79                                         |                                      | EXIT                                   | UPDATFLG               | 12 |
| 13 | MAINRTN1                                       | TC                                   | DSPLY81                                | 13                     |    |
| 14 |                                                | VLOAD                                | ABVAL                                  | 14                     |    |
| 15 |                                                |                                      | DELVEET3                               | 15                     |    |
| 16 |                                                |                                      |                                        | 16                     |    |
| 17 | STOVL                                          | DELVTPI                              | # DELTA V                              | 17                     |    |
| 18 |                                                | VPASS4                               |                                        | 18                     |    |
| 19 | VSU                                            | ABVAL                                |                                        | 19                     |    |
| 20 | STOVL                                          | VTPRIME                              |                                        | 20                     |    |
| 21 |                                                | DELVTPI                              | # DELTA V (FINAL) = V'T - VT           | 21                     |    |
| 22 |                                                | RACT3                                |                                        | 22                     |    |
| 23 |                                                |                                      |                                        | 23                     |    |
| 24 | PDVL                                           | CALL                                 |                                        | 24                     |    |
| 25 |                                                | VIPRIME                              |                                        | 25                     |    |
| 26 |                                                | PERIAP01                             | # GET PERIGEE ALTITUDE                 | 26                     |    |
| 27 |                                                |                                      |                                        | 27                     |    |
| 28 | CALL                                           |                                      |                                        | 28                     |    |
| 29 | STORE                                          | SHIFTR1                              |                                        | 29                     |    |
| 30 |                                                | POSTTPI                              |                                        | 30                     |    |
| 31 |                                                | BON                                  | SET                                    |                        | 31 |
| 32 |                                                |                                      |                                        | 32                     |    |
| 33 | DSPLY58                                        | FINALFLG                             |                                        | 33                     |    |
| 34 |                                                | DSPLY58                              |                                        | 34                     |    |
| 35 |                                                | UPDATFLG                             |                                        | 35                     |    |
| 36 | EXIT                                           |                                      |                                        | 36                     |    |
| 37 | CAF                                            | V06N58SR                             | # DISPLAY HP, DELTA V, DELTA V (FINAL) | 37                     |    |
| 38 | DSPLY81                                        | TC                                   | VNDSPLY                                | 38                     |    |
| 39 |                                                | CAF                                  | V06N81SR                               | # DISPLAY DELTA V (LV) | 39 |
| 40 |                                                | TC                                   | VNDSPLY                                |                        | 40 |
| 41 | TC                                             | INTPRET                              |                                        | 41                     |    |
| 42 |                                                | VLOAD                                |                                        | 42                     |    |
| 43 |                                                | CLEAR                                | XDELVFLG                               |                        | 43 |
| 44 |                                                |                                      |                                        | 44                     |    |
| 45 | STCALL                                         | DELVEET3                             |                                        | 45                     |    |
| 46 |                                                | DELVSIN                              |                                        | 46                     |    |
| 47 |                                                | VN1645                               | # DISPLAY TRKMKCNT, TTOGO, +MGA        | 47                     |    |
| 48 | BON                                            | GOTO                                 |                                        | 48                     |    |
| 49 |                                                | P39/79SW                             |                                        | 49                     |    |
| 50 |                                                | P39/P79B                             |                                        | 50                     |    |
| 51 |                                                |                                      |                                        | 51                     |    |
| 52 | RECYCLE                                        |                                      |                                        | 52                     |    |
| 53 |                                                |                                      |                                        | 53                     |    |
| 54 | # STABLE ORBIT MIDCOURSE PROGRAM (P39 AND P79) |                                      |                                        | 54                     |    |
| 55 | #                                              |                                      |                                        | 55                     |    |
| 56 | # MOD NO -1                                    | LOG SECTION - STABLE ORBIT - P38-P39 |                                        | 56                     |    |
| 57 | # MOD BY RUDNICKI.S                            | DATE 25JAN68                         |                                        | 57                     |    |
| 58 | #                                              |                                      |                                        | 58                     |    |
| 59 |                                                |                                      |                                        | 59                     |    |
| 60 |                                                |                                      |                                        | 60                     |    |

# STABLE ORBIT MIDCOURSE PROGRAM (P39 AND P79)

#

# MOD NO -1 LOG SECTION - STABLE ORBIT - P38-P39

# MOD BY RUDNICKI.S DATE 25JAN68

#

# FUNCTIONAL DESCRIPTION

#  
# P39 AND P79 CALCULATE THE REQUIRED DELTA V AND OTHER INITIAL  
# CONDITIONS REQUIRED BY THE AGC TO MAKE A MIDCOURSE CORRECTION  
# MANEUVER AFTER COMPLETING THE SOI MANEUVER BUT BEFORE MAKING  
# THE SOR MANEUVER.

# CALLING SEQUENCE

#  
# ASTRONAUT REQUEST THRU DSKY

# V37E39E IF THIS VEHICLE IS ACTIVE VEHICLE  
# V37E79E IF OTHER VEHICLE IS ACTIVE VEHICLE

# INPUT

#  
# (1) TPASS4 TIME OF INTERCEPT - SAVED FROM P38/P78  
# (2) TARGTIME TIME THAT PASSIVE VEHICLE IS AT INTERCEPT POINT -  
# SAVED FROM P38/P78

# OUTPUT

#  
# (1) TRKMKCNT NUMBER OF MARKS.  
# (2) TTOGO TIME TO GO  
# (3) +MGA MIDDLE GIMBAL ANGLE  
# (4) DELVLVC DELTA VELOCITY AT MID - LOCAL VERTICAL COORDINATES

# SUBROUTINES USED

#  
# AVFLAGA  
# AVFLAGP  
# LOADTIME  
# SELECTMU  
# PRECSET  
# S34/35.1  
# MAINRTNE

P39 TC BANKCALL  
CADR AVFLAGA # THIS VEHICLE ACTIVE

EXTEND  
DCA ATIGINC  
P79 TC P39/P79A  
TC BANKCALL  
CADR AVFLAGP # OTHER VEHICLE ACTIVE

EXTEND  
DCA PTIGINC  
P39/P79A DXCH KT # TIME TO PREPARE FOR BURN

TC BANKCALL  
CADR P20FLGON # SET UPDATFLG, TRACKFLG  
TC INTPRET

1412THE

|   |          |
|---|----------|
| # |          |
| # | PRECSET  |
| # | TIMETHET |
| # | S34/35.1 |

|         |        |          |                                         |
|---------|--------|----------|-----------------------------------------|
| PREC/TT | STQ    | DLOAD    |                                         |
|         |        | RTRN     |                                         |
|         |        | TIG      |                                         |
|         | STCALL | TDEC1    | # PRECISION UPDATE ACTIVE AND PASSIVE   |
|         |        | PRECSET  | #                                       |
|         | VLOAD  | VSR*     | VEHICLES TO TIG                         |
|         |        | RPASS3   |                                         |
|         |        | 0,2      |                                         |
|         | STODL  | RVEC     |                                         |
|         |        | CENTANG  |                                         |
|         | PUSH   | COS      |                                         |
|         | STODL  | CSTH     |                                         |
|         | SIN    | SET      |                                         |
|         |        | RVSW     |                                         |
|         | STOVL  | SNTH     |                                         |
|         |        | VPASS3   |                                         |
|         | VSR*   |          |                                         |
|         |        | 0,2      |                                         |
|         | STCALL | VVEC     | # GET TRANSFER TIME BASED ON CENTANG OF |
|         |        | TIMETHET | #                                       |
|         | CALL   |          | PASSIVE VEHICLE                         |
|         |        | S34/35.1 | # GET UNIT NORMAL                       |
|         | DLOAD  | GOTO     |                                         |
|         |        | T        |                                         |
|         |        | RTRN     |                                         |

```
#
CSMPREC
```

# STABLE\_ORBIT

# LEMPREC

INTRPVP STQ BOFF # PRECISION UPDATE PASSIVE VEHICLE TO  
RTRN # TDEC1

CALL  
OTHERV  
CSMPREC

GOTO

OTHERV CALL RTRN

GOTO LEMPREC  
RTRN

# .... VNDSPY ....  
# SUBROUTINES USED

#  
# BANKCALL  
# GOFLASH  
# GOTOPOOH

VNDSPY EXTEND # FLASH DISPLAY

QXCH RTRN  
TS VERBNOUN  
CA VERBNOUN  
TCR BANKCALL  
CADR GOFLASH  
TCF GOTOPOOH

TC RTRN # TERMINATE  
TCF -5 # PROCEED  
# RECYCLE

V06N33SR VN 0633  
V06N55SR VN 0655  
V04N06SR VN 0406  
V06N57SR VN 0657  
V06N34SR VN 0634  
V06N58SR VN 0658  
V06N81SR VN 0681

DECTWO OCT 2

# \*\*\* END OF KISSING .050 \*\*\*



```
##
BURN, BABY, BURN -- MASTER IGNITION ROUTINE
```

```
BANK 36
SETLOC P40S
BANK
EBANK= WHICH
COUNT* $$/P40
```

```
THE MASTER IGNITION ROUTINE IS DESIGNED FOR USE BY THE FOLLOWING LEM PROGRAMS: P12, P40, P42, P61, P63.
IT PERFORMS ALL FUNCTIONS IMMEDIATELY ASSOCIATED WITH APS OR DPS IGNITION: IN PARTICULAR, EVERYTHING LYING
BETWEEN THE PRE-IGNITION TIME CHECK -- ARE WE WITHIN 45 SECONDS OF TIG? -- AND TIG + 26 SECONDS, WHEN DPS
PROGRAMS THROTTLE UP.
```

```
VARIATIONS AMONG PROGRAMS ARE ACCOMODATED BY MEANS OF TABLES CONTAINING CONSTANTS (FOR AVEGEXIT, FOR
WAITLIST, FOR PINBALL) AND TCF INSTRUCTIONS. USERS PLACE THE ADRES OF THE HEAD OF THE APPROPRIATE TABLE
(OF P61TABLE FOR P61LM, FOR EXAMPLE) IN ERASABLE REGISTER 'WHICH' (E4). THE IGNITION ROUTINE THEN INDEXES BY
WHICH TO OBTAIN OR EXECUTE THE PROPER TABLE ENTRY. THE IGNITION ROUTINE IS INITIATED BY A TCF BURNBABY,
THROUGH BANKJUMP IF NECESSARY. THERE IS NO RETURN.
```

```
THE MASTER IGNITION ROUTINE WAS CONCEIVED AND EXECUTED, AND (NOTA BENE) IS MAINTAINED BY ADLER AND EYLES.
```

```
#
HONI SOIT QUI MAL Y PENSE
```

```
#

TABLES FOR THE IGNITION ROUTINE
```

```

```

```
NOLI SE TANGERE
```

|          |        |          |        |           |
|----------|--------|----------|--------|-----------|
| P12TABLE | VN     | 0674     | # (0)  |           |
|          | TCF    | ULLGNOT  | # (1)  |           |
|          | TCF    | COMFAIL3 | # (2)  |           |
|          | TCF    | GOCUTOFF | # (3)  |           |
|          | TCF    | TASKOVER | # (4)  |           |
|          | TCF    | P12SPOT  | # (5)  |           |
|          | DEC    | 0        | # (6)  | NO ULLAGE |
|          | EBANK= | WHICH    |        |           |
|          | 2CADR  | SERVEXIT | # (7)  |           |
|          | TCF    | DISPCHNG | # (11) |           |
|          | TCF    | WAITABIT | # (12) |           |
|          | TCF    | P12IGN   | # (13) |           |

|          |     |          |       |  |
|----------|-----|----------|-------|--|
| P40TABLE | VN  | 0640     | # (0) |  |
|          | TCF | ULLGNOT  | # (1) |  |
|          | TCF | COMFAIL4 | # (2) |  |
|          | TCF | GOPOST   | # (3) |  |
|          | TCF | TASKOVER | # (4) |  |
|          | TCF | P40SPOT  | # (5) |  |

|    |          |        |          |        |
|----|----------|--------|----------|--------|
| 1  |          |        |          |        |
| 2  |          | DEC    | 2240     | # (6)  |
| 3  |          | EBANK= | OMEGAQ   |        |
| 4  |          | 2CADR  | STEERING | # (7)  |
| 5  |          |        |          |        |
| 6  |          | TCF    | P40SJUNK | # (11) |
| 7  |          | TCF    | WAITABIT | # (12) |
| 8  |          | TCF    | P40IGN   | # (13) |
| 9  |          | TCF    | REP40ALM | # (14) |
| 10 |          |        |          |        |
| 11 | P41TABLE | TCF    | P41SPOT  | # (5)  |
| 12 |          | DEC    | -1       | # (6)  |
| 13 |          | EBANK= | OMEGAQ   |        |
| 14 |          | 2CADR  | CALCN85  | # (7)  |
| 15 |          |        |          |        |
| 16 |          | TCF    | COMMON   | # (11) |
| 17 |          | TCF    | TIGTASK  | # (12) |
| 18 |          |        |          |        |
| 19 | P42TABLE | VN     | 0640     | # (0)  |
| 20 |          | TCF    | WANTAPS  | # (1)  |
| 21 |          | TCF    | COMFAIL4 | # (2)  |
| 22 |          | TCF    | GOPOST   | # (3)  |
| 23 |          | TCF    | TASKOVER | # (4)  |
| 24 |          | TCF    | P42SPOT  | # (5)  |
| 25 |          | DEC    | 2640     | # (6)  |
| 26 |          | EBANK= | OMEGAQ   |        |
| 27 |          | 2CADR  | STEERING | # (7)  |
| 28 |          |        |          |        |
| 29 |          | TCF    | P40SJUNK | # (11) |
| 30 |          | TCF    | WAITABIT | # (12) |
| 31 |          | TCF    | P42IGN   | # (13) |
| 32 |          | TCF    | P42STAGE | # (14) |
| 33 |          |        |          |        |
| 34 | P63TABLE | VN     | 0662     | # (0)  |
| 35 |          | TCF    | ULLGNOT  | # (1)  |
| 36 |          | TCF    | COMFAIL3 | # (2)  |
| 37 |          | TCF    | V99RECYC | # (3)  |
| 38 |          | TCF    | TASKOVER | # (4)  |
| 39 |          | TCF    | P63SPOT  | # (5)  |
| 40 |          | DEC    | 2240     | # (6)  |
| 41 |          | EBANK= | WHICH    |        |
| 42 |          | 2CADR  | SERVEXIT | # (7)  |
| 43 |          |        |          |        |
| 44 |          | TCF    | DISPCHNG | # (11) |
| 45 |          | TCF    | WAITABIT | # (12) |
| 46 |          |        |          |        |
| 47 |          |        |          |        |
| 48 |          |        |          |        |
| 49 |          |        |          |        |
| 50 |          |        |          |        |
| 51 |          |        |          |        |
| 52 |          |        |          |        |
| 53 |          |        |          |        |
| 54 |          |        |          |        |
| 55 |          |        |          |        |
| 56 |          |        |          |        |
| 57 |          |        |          |        |
| 58 |          |        |          |        |
| 59 |          |        |          |        |
| 60 |          |        |          |        |

```
1
2 TCF P63IGN # (13)
3
4 ABRTABLE VN 0663 # (0)
5 TCF ULLGNOT # (1)
6 TCF COMFAIL3 # (2)
7 TCF GOCUTOFF # (3)
8 TCF TASKOVER # (4)
9 NOOP # (5)
10 NOOP # (6)
11 NOOP # (7)
12 NOOP
13 TCF DISPCHNG # (11)
14 TCF WAITABIT # (12)
15 TCF ABRTIGN # (13)
16
17 # *****
18 # GENERAL PURPOSE IGNITION ROUTINES
19 # *****
20
21 BURNBABY TC PHASCHNG # GROUP 4 RESTARTS HERE
22 OCT 04024
23
24 CAF ZERO # EXTIRPATE JUNK LEFT IN DVTOTAL
25 TS DVTOTAL
26 TS DVTOTAL +1
27
28 TC BANKCALL # P40AUTO MUST BE BANKCALLED EVEN FROM ITS
29 CADR P40AUTO # OWN BANK TO SET UP RETURN PROPERLY
30
31 B*RNB*B* EXTEND
32 DCA TIG # STORE NOMINAL TIG FOR OBLATENESS COMP.
33 DXCH GOBLTIME # AND FOR P70 OR P71.
34
35 INHINT
36 TC IBNKCALL
37 CADR ENGINOF3
38 RELINT
39
40 INDEX WHICH
41 TCF 5
42
43 P42SPOT = P40SPOT # (5)
44 P12SPOT = P40SPOT # (5)
45 P63SPOT = P41SPOT # (5) IN P63 CLOKTASK ALREADY GOING
46 P40SPOT CS CNTDNDEX # (5)
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

|          |        |          |                                          |
|----------|--------|----------|------------------------------------------|
|          | TC     | BANKCALL | # MUST BE BANKCALLED FOR GENERALIZED     |
|          | CADR   | STCLOK2  | # RETURN                                 |
| P41SPOT  | TC     | INTPRET  | # (5)                                    |
|          | DLOAD  | DSU      |                                          |
|          |        | TIG      |                                          |
|          | STCALL | D29.9SEC |                                          |
|          |        | TDEC1    |                                          |
|          |        | INITCDUW |                                          |
|          | BOFF   | CALL     |                                          |
|          |        | MUNFLAG  |                                          |
|          |        | GOMIDAV  |                                          |
|          | VLOAD  | CSMPREC  |                                          |
|          |        | MXV      |                                          |
|          |        | VATT1    |                                          |
|          |        | REFSMMAT |                                          |
|          | VSR1   |          |                                          |
|          | STOVL  | V(CSM)   | # CSM VELOCITY -- M/CS*2(7)              |
|          | VSL4   | RATT1    |                                          |
|          |        | MXV      |                                          |
|          |        | REFSMMAT |                                          |
|          | STCALL | R(CSM)   | # CSM POSITION -- M*2(24)                |
|          |        | MUNGRAV  |                                          |
|          | STODL  | G(CSM)   | # CSM GRAVITY VEC. -- M/CS*2(7)          |
|          |        | TAT      |                                          |
|          | STORE  | TDEC1    | # RELOAD TDEC1 FOR MIDTOAV.              |
| GOMIDAV  | CALRB  |          |                                          |
|          | TCF    | MIDTOAV1 |                                          |
|          |        | CALLT-35 | # MADE IT IN TIME.                       |
|          | EXTEND |          | # TIG WAS SLIPPED, SO RESET TIG TO 29.9  |
|          | DCA    | PIPTIME1 | # SECONDS AFTER THE TIME TO WHICH WE DID |
|          | DXCH   | TIG      | # INTEGRATE.                             |
|          | EXTEND |          |                                          |
|          | DCA    | D29.9SEC |                                          |
|          | DAS    | TIG      |                                          |
|          | DXCH   | MPAC     |                                          |
| CALLT-35 | DXCH   | SAVET-30 | # DELTA-T UNTIL TIG-30                   |
|          | EXTEND |          |                                          |
|          | DCS    | 5SECDP   |                                          |
|          | DAS    | SAVET-30 | # DELTA-T UNTIL TIG-35                   |
|          | EXTEND |          |                                          |
|          | DCA    | SAVET-30 |                                          |
|          | TC     | LONGCALL |                                          |
|          | EBANK= | TTOGO    |                                          |
|          | 2CADR  | TIG-35   |                                          |
|          | TC     | PHASCHNG |                                          |
|          | OCT    | 20254    | # 4.25SPOT FOR TIG-35 RESTART.           |

```
1
2 TC CHECKMM
3 DEC 63
4 TCF ENDOFJOB # NOT P63
5 CS CNTDNDEX # P63 CAN START DISPLAYING NOW.
6 TS DISPDEX
7 TC INTPRET
8 VLOAD ABVAL
9 VN1
10 STORE ABVEL # INITIALIZE ABVEL FOR P63 DISPLAY
11 EXIT
12 TCF ENDOFJOB
13
14 # *****
15
16 TIG-35 CAF 5SEC
17 TC TWIDDLE
18 ADRES TIG-30
19
20 TC PHASCHNG
21 OCT 40154 # 4.15SPOT FOR TIG-30 RESTART
22
23 CS BLANKDEX # BLANK DSKY FOR 5 SECONDS
24 TS DISPDEX
25
26 INDEX WHICH
27 CS 6 # CHECK ULLAGE TIME.
28 EXTEND
29 BZMF TASKOVER
30 CAF 4.9SEC # SET UP TASK TO RESTORE DISPLAY AT TIG-30
31 TC TWIDDLE
32 ADRES TIG-30.1
33
34 CAF PRI017 # A NEGATIVE ULLAGE TIME INDICATES P41, IN
35 TC NOVAC # WHICH CASE WE HAVE TO SET UP A JOB TO
36 EBANK= TTOGO # BLANK THE DSKY FOR FIVE SECONDS, SINCE
37 2CADR P41BLANK # CLOKJOB IS NOT RUNNING DURING P41.
38
39 TCF TASKOVER
40
41 P41BLANK TC BANKCALL # BLANK DSKY.
42 CADR CLEANDSP
43 TCF ENDOFJOB
44
45 TIG-30.1 CAF PRI017 # SET UP JOB TO RESTORE DISPLAY AT TIG-30
46 TC NOVAC
47 EBANK= TTOGO
48 2CADR TIG-30A
49
50 TCF TASKOVER
51
52
53
54
55
56
57
58
59
60
```

```
1 TIG-30A CAF V16N85B
2 TC BANKCALL # RESTORE DISPLAY.
3 CADR REGODSP # REGODSP DOES A TCF ENDOFJOB
4
5 # *****
6
7 TIG-30 CAF S24.9SEC
8 TC TWIDDLE
9 ADRES TIG-5
10
11 CS CNTDNDEX # START UP CLOKTASK AGAIN
12 TS DISPDEX
13
14 INDEX WHICH # PICK UP APPROPRIATE ULLAGE -- ON TIME
15 CA 6 # WAS CAF --- RSB 2009.
16 EXTEND
17 BZMF ULLGNOT # DON'T SET UP ULLAGE IF DT IS NEG OR ZERO
18 TS SAVET-30 # SAVE DELTA-T FOR RESTART
19 TC TWIDDLE
20 ADRES ULLGTASK
21
22
23 CA THREE # RESTART PROTECT ULLGTASK (1.3SPOT)
24 TS L
25 CS THREE
26 DXCH -PHASE1
27 CS TIME1
28 TS TBASE1
29
30 INDEX WHICH
31 TCF 1
32
33 WANTAPS CS FLGWRD10 # (1) FOR P42 ENSURE APSFLAG IS SET. IF IT
34 MASK APSFLBIT # WASN'T SET, DAP WILL BE INITIALIZED TO
35 ADS FLGWRD10 # ASCENT VALUES BY 1/ACCS IN 2 SECONDS.
36
37 ULLGNOT EXTEND
38 INDEX WHICH # (1)
39 DCA 7 # LOAD AVEGEXIT WITH APPROPRIATE 2CADR
40 DXCH AVEGEXIT
41
42 CAF TWO # 4.2SPOT RESTARTS IMMEDIATELY AT RED04.2
43 TS L
44 CS TWO # AND ALSO AT TIG-5 AT THE CORRECT TIME.
45 DXCH -PHASE4
46
47 CS TIME1
48 TS TBASE4 # SET TBASE4 FOR TIG-5 RESTART
49
50 RED02.17 EXTEND
```

```
1
2 DCA NEG0 # CLEAR OUT GROUP 2 SO LAMBERT CAN START
3 DXCH -PHASE2 # IF NEEDED.
```

```
4
5 RED04.2 CCS PHASE5 # IF SERVICER GOING?
6 TCF TASKOVER # YES, DON'T START IT UP AGAIN.
```

```
7
8 TC POSTJUMP
9 CADR PREREAD # PREREAD END THIS TASK
```

```
10
11 # *****
```

```
12
13 ULLGTASK TC ONULLAGE # THIS COMES AT TIG-7.5 OR TIG-3.5
14 TC PHASCHNG
15 OCT 1
16 TCF TASKOVER
```

```
17
18 # *****
```

```
19
20 TIG-5 EXTEND
21 DCA NEG0 # INSURE THAT GROUP 3 IS INACTIVE.
22 DXCH -PHASE3
```

```
23
24 CAF 5SEC
25 TC TWIDDLE
26 ADRES TIG-0
```

```
27
28 TC DOWNFLAG # RESET IGNFLAG AND ASINFLAG
29 ADRES IGNFLAG # FOR LIGHT-UP LOGIC.
30 TC DOWNFLAG
31 ADRES ASTNFLAG
```

```
32
33 INDEX WHICH
34 TCF 11
```

```
35
36 P40SJUNK CCS PHASE3 # (11) P40 AND P42. S40.13 IN PROGRESS?
37 TCF DISPCHNG # YES
```

```
38
39 CAF PRI020
40 TC FINDVAC
41 EBANK= TTOGO
42 2CADR S40.13
```

```
43
44 TC PHASCHNG # 3.5SPOT FOR S40.13
45 OCT 00053
```

```
46 DISPCHNG CS VB99DEX # (11)
47 TS DISPDEX
```

```
1 COMMON TC PHASCHNG # RESTART TIG-0 (4.7SPOT)
2 OCT 40074
3 TCF TASKOVER
4
5 # *****
6
7 TIG-0 CS FLAGWRD7 # SET IGNFLAG SINCE TIG HAS ARRIVED
8 MASK IGNFLBIT
9 ADS FLAGWRD7
10
11
12 TC CHECKMM # IN P63 CASE, THROTTLE-UP IS ZOOMTIME
13 DEC 63 # AFTER NOMINAL IGNITION, NOT ACTUAL
14 TCF IGNYET?
15 CA ZOOMTIME
16 TC WAITLIST
17 EBANK= DVCNTR
18 2CADR P63ZOOM
19
20 TC 2PHSCHNG
21 OCT 40033
22
23 OCT 05014
24 OCT 77777
25
26 IGNYET? CAF ASTNBIT # CHECK ASTNFLAG: HAS ASTRONAUT RESPONDED
27 MASK FLAGWRD7 # TO OUR ENGINE ENABLE REQUEST?
28 EXTEND
29 INDEX WHICH
30 BZF 12 # BRANCH IF HE HAS NOT RESPONDED YET
31
32 IGNITION CS FLAGWRD5 # INSURE ENGONFLG IS SET.
33 MASK ENGONBIT
34 ADS FLAGWRD5
35 CS PRI030 # TURN ON THE ENGINE.
36 EXTEND
37 RAND DSALMOUT
38 AD BIT13
39 EXTEND
40 WRITE DSALMOUT
41 EXTEND
42 DCA TIME2 # SET TEVENT FOR DOWNLINK
43 DXCH TEVENT
44
45 EXTEND # UPDATE TIG USING TGO FROM S40.13
46 DCA TGO
47 DXCH TIG
48 EXTEND
49 DCA TIME2
50 DAS TIG
51
52
53
54
55
56
57
58
59
60
```



|    |         |        |          |                                           |
|----|---------|--------|----------|-------------------------------------------|
| 1  |         |        |          |                                           |
| 2  |         | CS     | FLUNDBIT | # PERMIT GUIDANCE LOOP DISPLAYS           |
| 3  |         | MASK   | FLAGWRD8 |                                           |
| 4  |         | TS     | FLAGWRD8 |                                           |
| 5  |         |        |          |                                           |
| 6  |         | INDEX  | WHICH    |                                           |
| 7  |         | TCF    | 13       |                                           |
| 8  |         |        |          |                                           |
| 9  | P63IGN  | EXTEND |          | # (13) INITIATE BURN DISPLAYS             |
| 10 |         | DCA    | DSP2CADR |                                           |
| 11 |         | DXCH   | AVGEXIT  |                                           |
| 12 |         |        |          |                                           |
| 13 |         | CA     | Z        | # ASSASSINATE CLOKTASK                    |
| 14 |         | TS     | DISPDEX  |                                           |
| 15 |         |        |          |                                           |
| 16 |         | CS     | FLAGWRD9 | # SET FLAG FOR P70-P71                    |
| 17 |         | MASK   | LETABBIT |                                           |
| 18 |         | ADS    | FLAGWRD9 |                                           |
| 19 |         |        |          |                                           |
| 20 |         | CS     | FLAGWRD7 | # SET SWANDISP TO ENABLE R10.             |
| 21 |         | MASK   | SWANDBIT |                                           |
| 22 |         | ADS    | FLAGWRD7 |                                           |
| 23 |         |        |          |                                           |
| 24 |         | CS     | PULSES   | # MAKE SURE DAP IS NOT IN MINIMUM-IMPULSE |
| 25 |         | MASK   | DAPBOOLS | # MODE, IN CASE OF SWITCH TO P66          |
| 26 |         | TS     | DAPBOOLS |                                           |
| 27 |         |        |          |                                           |
| 28 |         | EXTEND |          | # INITIALIZE TIG FOR P70 AND P71.         |
| 29 |         | DCA    | TIME2    |                                           |
| 30 |         | DXCH   | TIG      |                                           |
| 31 |         |        |          |                                           |
| 32 |         | CAF    | ZERO     | # INITIALIZE WCHPHASE, AND FLPASS0        |
| 33 |         | TS     | WCHPHASE |                                           |
| 34 |         | TS     | WCHPHOLD | # ALSO WHCPHOLD                           |
| 35 |         | CA     | TWO      |                                           |
| 36 |         | TS     | FLPASS0  |                                           |
| 37 |         |        |          |                                           |
| 38 |         | TCF    | P42IGN   |                                           |
| 39 | P40IGN  | CS     | FLAGWRD5 | # (13)                                    |
| 40 |         | MASK   | NOTHRBIT |                                           |
| 41 |         | EXTEND |          |                                           |
| 42 |         | BZF    | P42IGN   |                                           |
| 43 |         | CA     | ZOOMTIME |                                           |
| 44 |         | TC     | WAITLIST |                                           |
| 45 |         | EBANK= | DVCNTR   |                                           |
| 46 |         | 2CADR  | P40ZOOM  |                                           |
| 47 |         |        |          |                                           |
| 48 | P63IGN1 | TC     | 2PHSCHNG |                                           |
| 49 |         | OCT    | 40033    | # 3.3SPOT FOR ZOOM RESTART.               |
| 50 |         | OCT    | 05014    | # TYPE C RESTARTS HERE IMMEDIATELY        |
| 51 |         | OCT    | 77777    |                                           |
| 52 |         |        |          |                                           |
| 53 |         |        |          |                                           |
| 54 |         |        |          |                                           |
| 55 |         |        |          |                                           |
| 56 |         |        |          |                                           |
| 57 |         |        |          |                                           |
| 58 |         |        |          |                                           |
| 59 |         |        |          |                                           |
| 60 |         |        |          |                                           |

```
1
2 P12IGN TCF P42IGN
3 CAF EBANK6
4 TS EBANK
5 EBANK= AOSQ
6
7 CA IGNAOSQ # INITIALIZE DAP BIAS ACCELERATION
8 TS AOSQ # ESTIMATES AT P12 IGNITION.
9 CA IGNAOSR
10 TS AOSR
11
12 CAF EBANK7
13 TS EBANK
14 EBANK= DVCNTR
15
16 ABRTIGN CA Z # (13) KILL CLOKTASK
17 TS DISPDEX
18
19 EXTEND # CONNECT ASCENT GYIDANCE TO SERVICER.
20 DCA ATMAGADR
21 DXCH AVGEXIT
22
23 CS FLAGWRD7 # ENABLE R10.
24 MASK SWANDBIT
25 ADS FLAGWRD7
26
27 P42IGN CS DRIFTBIT # ENSURE THAT POWERED-FLIGHT SWITCHING
28 MASK DAPBOOLS # CURVES ARE USED.
29 TS DAPBOOLS
30 CAF IMPULBIT # EXAMINE IMPULSE SWITCH
31 MASK FLAGWRD2
32 CCS A
33 TCF IMPLBURN
34
35 DVMONCON TC DOWNFLAG
36 ADRES IGNFLAG # CONNECT DVMON
37 TC DOWNFLAG
38 ADRES ASTNFLAG
39 TC DOWNFLAG
40 ADRES IDLEFLAG
41
42 TC PHASCHNG
43 OCT 40054
44
45 TC FIXDELAY # TURN ULLAGE OFF HALF A SECOND AFTER
46 DEC 50 # LIGHT UP.
47
48 ULLAGOFF TC NOULLAGE
49
50 WAITABIT EXTEND # KILL GROUP 4
51 DCA NEG0
52
53
54
55
56
57
58
59
60
```

|    |          |        |                  |    |
|----|----------|--------|------------------|----|
| 1  |          |        |                  | 1  |
| 2  |          | DXCH   | -PHASE4          | 2  |
| 3  |          |        |                  | 3  |
| 4  |          | TCF    | TASKOVER         | 4  |
| 5  |          |        |                  | 5  |
| 6  | TIGTASK  | TC     | POSTJUMP # (12)  | 6  |
| 7  |          | CADR   | TIGTASK1         | 7  |
| 8  |          |        |                  | 8  |
| 9  | #        | *****  |                  | 9  |
| 10 |          |        |                  | 10 |
| 11 |          | BANK   | 31               | 11 |
| 12 |          | SETLOC | P40S3            | 12 |
| 13 |          |        |                  | 13 |
| 14 |          | BANK   |                  | 14 |
| 15 |          | COUNT* | \$\$/P40         | 15 |
| 16 | TIGTASK1 | CAF    | PRI016           | 16 |
| 17 |          | TC     | NOVAC            | 17 |
| 18 |          | EBANK= | TRKMKCNT         | 18 |
| 19 |          | 2CADR  | TIGNOW           | 19 |
| 20 |          |        |                  | 20 |
| 21 |          | TC     | PHASCHNG         | 21 |
| 22 |          | OCT    | 6 # KILL GROUP 6 | 22 |
| 23 |          |        |                  | 23 |
| 24 |          | TCF    | TASKOVER         | 24 |
| 25 |          |        |                  | 25 |
| 26 | #        | *****  |                  | 26 |
| 27 |          |        |                  | 27 |
| 28 | P63ZOOM  | EXTEND |                  | 28 |
| 29 |          | DCA    | LUNLANAD         | 29 |
| 30 |          | DXCH   | AVEGEXIT         | 30 |
| 31 |          |        |                  | 31 |
| 32 |          | TC     | IBNKCALL         | 32 |
| 33 |          | CADR   | FLATOUT          | 33 |
| 34 |          | TCF    | P40ZOOMA         | 34 |
| 35 |          |        |                  | 35 |
| 36 | P40ZOOM  | CAF    | BIT13            | 36 |
| 37 |          | TS     | THRUST           | 37 |
| 38 |          | CAF    | BIT4             | 38 |
| 39 |          |        |                  | 39 |
| 40 |          | EXTEND |                  | 40 |
| 41 |          | WOR    | CHAN14           | 41 |
| 42 |          |        |                  | 42 |
| 43 | P40ZOOMA | TC     | PHASCHNG         | 43 |
| 44 |          | OCT    | 3                | 44 |
| 45 |          | TCF    | TASKOVER         | 45 |
| 46 |          |        |                  | 46 |
| 47 |          | EBANK= | DVCNTR           | 47 |
| 48 | LUNLANAD | 2CADR  | LUNLAND          | 48 |
| 49 |          |        |                  | 49 |
| 50 |          |        |                  | 50 |
| 51 |          |        |                  | 51 |
| 52 |          |        |                  | 52 |
| 53 |          |        |                  | 53 |
| 54 |          |        |                  | 54 |
| 55 |          |        |                  | 55 |
| 56 |          |        |                  | 56 |
| 57 |          |        |                  | 57 |
| 58 |          |        |                  | 58 |
| 59 |          |        |                  | 59 |
| 60 |          |        |                  | 60 |

```
1 ZOOM = P40ZOOMA
2 BANK 36
3 SETLOC P40S
4 BANK
5 COUNT* $$/P40
6
7 # *****
8
9 COMFAIL TC UPFLAG # (15)
10 ADRES IDLEFLAG
11 TC UPFLAG # SET FLAG TO SUPPRESS CONFLICTING DISPLAY
12 ADRES FLUNDISP
13 CAF FOUR # RESET DVMON
14 TS DVCNTR
15 CCS PHASE6 # CLOCKTASK ACTIVE?
16 TCF +3 # YES
17 TC BANKCALL # OTHERWISE, START IT UP
18
19 +3 CADR STCLOK1
20 CS VB97DEX
21 TS DISPDEX
22 TC PHASCHNG # TURN OFF GROUP 4.
23 OCT 00004
24 TCF ENDOFJOB
25
26 COMFAIL1 INDEX WHICH
27 TCF 2
28
29 COMFAIL3 CA Z # (15) KILL CLOKTASK USING Z
30 TCF +2
31
32 COMFAIL4 CS CNTDNDEX
33 TS DISPDEX
34
35 TC DOWNFLAG # RECONNECT DV MONITOR
36 ADRES IDLEFLAG
37 TC DOWNFLAG # PERMIT GUIDANCE LOOP DISPLAYS
38 ADRES FLUNDISP
39 TCF ENDOFJOB
40
41 COMFAIL2 TC PHASCHNG # KILL ZOOM RESTART PROTECTION
42 OCT 00003
43
44 INHINT
45 TC KILLTASK # KILL ZOOM IN CASE IT'S STILL TO COME
46 CADR ZOOM
47 TC IBNKCALL # COMMAND ENGINE OFF
48 CADR ENGINOF4
49 TC UPFLAG # SET THE DRIFT BIT FOR THE DAP.
50 ADRES DRIFTDFL
51
52
53
54
55
56
57
58
59
60
```

|    |          |                                     |          |                                            |    |
|----|----------|-------------------------------------|----------|--------------------------------------------|----|
| 1  |          |                                     |          |                                            | 1  |
| 2  |          | TC                                  | INVFLAG  | # USE OTHER RCS SYSTEM                     | 2  |
| 3  |          | ADRES                               | AORBTFLG |                                            | 3  |
| 4  |          | TC                                  | UPFLAG   | # TURN ON ULLAGE                           | 4  |
| 5  |          | ADRES                               | ULLAGFLG |                                            | 5  |
| 6  |          | CAF                                 | BIT1     |                                            | 6  |
| 7  |          | INHINT                              |          |                                            | 7  |
| 8  |          | TC                                  | TWIDDLE  |                                            | 8  |
| 9  |          | ADRES                               | TIG-5    |                                            | 9  |
| 10 |          | TCF                                 | ENDOFJOB |                                            | 10 |
| 11 |          |                                     |          |                                            | 11 |
| 12 | #        | *****                               |          |                                            | 12 |
| 13 | #        | SUBROUTINES OF THE IGNITION ROUTINE |          |                                            | 13 |
| 14 | #        | *****                               |          |                                            | 14 |
| 15 |          |                                     |          |                                            | 15 |
| 16 | INVFLAG  | CA                                  | Q        |                                            | 16 |
| 17 |          | TC                                  | DEBIT    |                                            | 17 |
| 18 |          | COM                                 |          |                                            | 18 |
| 19 |          | EXTEND                              |          |                                            | 19 |
| 20 |          | RXOR                                | LCHAN    |                                            | 20 |
| 21 |          | TCF                                 | COMFLAG  |                                            | 21 |
| 22 |          |                                     |          |                                            | 22 |
| 23 | #        | *****                               |          |                                            | 23 |
| 24 |          |                                     |          |                                            | 24 |
| 25 | NOULLAGE | CS                                  | ULLAGER  | # MUST BE CALLED IN A TASK OR UNDER INHINT | 25 |
| 26 |          | MASK                                | DAPBOOLS |                                            | 26 |
| 27 |          | TS                                  | DAPBOOLS |                                            | 27 |
| 28 |          | TC                                  | Q        |                                            | 28 |
| 29 |          |                                     |          |                                            | 29 |
| 30 | #        | *****                               |          |                                            | 30 |
| 31 |          |                                     |          |                                            | 31 |
| 32 | ONULLAGE | CS                                  | DAPBOOLS | # TURN ON ULLAGE. MUST BE CALLED IN        | 32 |
| 33 |          | MASK                                | ULLAGER  | # A TASK OR WHILE INHINTED.                | 33 |
| 34 |          | ADS                                 | DAPBOOLS |                                            | 34 |
| 35 |          | TC                                  | Q        |                                            | 35 |
| 36 |          |                                     |          |                                            | 36 |
| 37 | #        | *****                               |          |                                            | 37 |
| 38 |          |                                     |          |                                            | 38 |
| 39 | STCLOK1  | CA                                  | ZERO     | # THIS ROUTINE STARTS THE COUNT-DOWN       | 39 |
| 40 | STCLOK2  | TS                                  | DISPDEX  | # (CLOKTASK AND CLOKJOB). SETTING          | 40 |
| 41 | STCLOK3  | TC                                  | MAKECADR | # SETTING DISPDEX POSITIVE KILLS IT.       | 41 |
| 42 |          | TS                                  | TBASE4   | # RETURN SAVE (NOT FOR RESTARTS).          | 42 |
| 43 |          | EXTEND                              |          |                                            | 43 |
| 44 |          | DCA                                 | TIG      |                                            | 44 |
| 45 |          | DXCH                                | MPAC     |                                            | 45 |
| 46 |          | EXTEND                              |          |                                            | 46 |
| 47 |          | DCS                                 | TIME2    |                                            | 47 |
| 48 |          |                                     |          |                                            | 48 |
| 49 |          |                                     |          |                                            | 49 |
| 50 |          |                                     |          |                                            | 50 |
| 51 |          |                                     |          |                                            | 51 |
| 52 |          |                                     |          |                                            | 52 |
| 53 |          |                                     |          |                                            | 53 |
| 54 |          |                                     |          |                                            | 54 |
| 55 |          |                                     |          |                                            | 55 |
| 56 |          |                                     |          |                                            | 56 |
| 57 |          |                                     |          |                                            | 57 |
| 58 |          |                                     |          |                                            | 58 |
| 59 |          |                                     |          |                                            | 59 |
| 60 |          |                                     |          |                                            | 60 |

|    |          |        |          |                                             |
|----|----------|--------|----------|---------------------------------------------|
| 1  |          |        |          |                                             |
| 2  |          | DAS    | MPAC     | # HAVE TIG -- TIME2, UNDOUBTEDLY A + NUMBER |
| 3  |          | TC     | TPAGREE  | # POSITIVE, SINCE WE PASSED THE             |
| 4  |          | CAF    | 1SEC     | # 45 SECOND CHECK.                          |
| 5  |          | TS     | Q        |                                             |
| 6  |          | DXCH   | MPAC     |                                             |
| 7  |          | MASK   | LOW5     | # RESTRICT MAGNITUDE OF NUMBER IN A         |
| 8  |          | EXTEND |          |                                             |
| 9  |          | DV     | Q        |                                             |
| 10 |          | CA     | L        | # GET REMAINDER                             |
| 11 |          | AD     | TWO      |                                             |
| 12 |          | INHINT |          |                                             |
| 13 |          | TC     | TWIDDLE  |                                             |
| 14 |          | ADRES  | CLOKTASK |                                             |
| 15 |          | TC     | 2PHSCHNG |                                             |
| 16 |          | OCT    | 40036    | # 6.3SPOT FOR CLOKTASK                      |
| 17 |          | OCT    | 05024    |                                             |
| 18 |          | OCT    | 13000    |                                             |
| 19 |          |        |          |                                             |
| 20 |          | CA     | TBASE4   |                                             |
| 21 |          | TC     | BANKJUMP |                                             |
| 22 |          |        |          |                                             |
| 23 | CLOKTASK | CS     | TIME1    | # SET TBASE6 FOR GROUP 6 RESTART            |
| 24 |          | TS     | TBASE6   |                                             |
| 25 |          |        |          |                                             |
| 26 |          | CCS    | DISPDEX  |                                             |
| 27 |          | TCF    | KILLCLOK |                                             |
| 28 |          | NOOP   |          |                                             |
| 29 |          | CAF    | PRI027   |                                             |
| 30 |          | TC     | NOVAC    |                                             |
| 31 |          | EBANK= | TTOGO    |                                             |
| 32 |          | 2CADR  | CLOKJOB  |                                             |
| 33 |          |        |          |                                             |
| 34 |          | TC     | FIXDELAY | # WAIT A SECOND BEFORE STARTING OVER        |
| 35 |          | DEC    | 100      |                                             |
| 36 |          | TCF    | CLOKTASK |                                             |
| 37 |          |        |          |                                             |
| 38 | KILLCLOK | EXTEND |          | # KILL RESTART                              |
| 39 |          | DCA    | NEGO     |                                             |
| 40 |          | DXCH   | -PHASE6  |                                             |
| 41 |          | TCF    | TASKOVER |                                             |
| 42 |          |        |          |                                             |
| 43 | CLOKJOB  | EXTEND |          |                                             |
| 44 |          | DCS    | TIG      |                                             |
| 45 |          | DXCH   | TTOGO    |                                             |
| 46 |          | EXTEND |          |                                             |
| 47 |          |        |          |                                             |
| 48 |          |        |          |                                             |
| 49 |          |        |          |                                             |
| 50 |          |        |          |                                             |
| 51 |          |        |          |                                             |
| 52 |          |        |          |                                             |
| 53 |          |        |          |                                             |
| 54 |          |        |          |                                             |
| 55 |          |        |          |                                             |
| 56 |          |        |          |                                             |
| 57 |          |        |          |                                             |
| 58 |          |        |          |                                             |
| 59 |          |        |          |                                             |
| 60 |          |        |          |                                             |

```
1
2 DCA TIME2
3 DAS TTOGO
4
5 INHINT
6 CCS DISPDEX # IF DISPDEX HAS BEEN SET POSITIVE BY A
7 TCF ENDOFJOB # TASK OR A HIGHER PRIORITY JOB SINCE THE
8
9 TCF ENDOFJOB # LAST CLOKTASK, AVOID USING IT AS AN
10 COM
11 RELINT # INDEX.
12 # ***** DISPDEX MUST NEVER B -0 *****
13
14 INDEX A
15 TCF DISPNOT -1 # (-1 DUE TO EFFECT OF CCS)
16
17 VB97DEX = OCT35 # NEGATIVE OF THIS IS PROPER FOR DISPDEX
18
19 -35 CS ZERO # INDICATE VERB 97 PASTE
20
21 TS NVWORD1
22 CA NVWORD +2 # NVWORD+2 CONTAINS V06 & APPROPRIATE NOUN
23 TC BANKCALL
24
25 CADR CLOCPLAY
26 TCF STOPCLOK # TERMINATE CLOKTASK ON THE WAY TO POOH
27 TCF COMFAIL1
28 TCF COMFAIL2
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

# THIS DISPLAY IS CALLED VIA ASTNCLOK  
# IT IS PRIMARILY USED BY THE CREW IN P63  
# TO RESET HIS EVENT TIMER TO AGREE WITH  
# TIG.

```
CAF V06N61
TC BANKCALL
CADR REFLASH
TCF STOPCLOK
TCF ASTNRETN
TCF -6

CNTDNDEX = LOW4 # OCT17: NEGATIVE PROPER FOR DISPDEX

-17 INDEX WHICH # THIS DISPLAY COMES UP AT ONE SECOND
WAS CAF --- RSB 2009
CA 0 # INTERVALS. IT IS NORMALLY OPERATED
TC BANKCALL # BETWEEN TIG-30 SECONDS AND TIG-5 SECONDS
CADR REGODSP # REGODSP DOES ITS OWN TCF ENDOFJOB

VB99DEX = ELEVEN # OCT13: NEGATIVE PROPER FOR DISPDEX

V99RECYC EQUALS

-13 CS BIT9 # INDICATE VERB 99 PASTE
TS NVWORD1
INDEX WHICH # THIS IS THE "PLEASE ENABLE ENGINE"
WAS CAF --- RSB 2004
CA 0 # DISPLAY; IT IS INITIATED AT TIG-5 SEC.
TC BANKCALL # THE DISPLAY IS A V99NXX, WHERE XX IS
CADR CLOCPLAY # NOUN THAT HAD PREVIOUSLY BEEN DISPLAYED
TCF STOPCLOK # TERMINATE GOTOPOOH TURNS OFF ULLAGE.
TCF *PROCEED
TCF *ENTER
```

```
1 BLANKDEX = TWO # NEGATIVE OF THIS IS PROPER FOR DISPDEX
2
3
4 -2 TC BANKCALL # BLANK DSKY. THE DSKY IS BLANKED FOR
5 CADR CLEANDSP # 5 SECONDS AT TIG-35 TO INDICATE THAT
6 DISPNOT TCF ENDOFJOB # AVERAGE G IS STARTING.
7
8 STOPCLOK TC NULLCLOK # STOP CLOKTASK & TURN OFF ULLAGE ON THE
9 TCF GOTOP00H # WAY TO P00 (GOTOP00H RELINTS)
10
11 NULLCLOK INHINT
12 EXTEND
13
14 QXCH P40/RET
15 TC NOULLAGE # TURN OFF ULLAGE ...
16 TC KILLTASK # DON'T LET IT COME ON, EITHER ...
17
18 CADR ULLGTASK
19 TC PHASCHNG #
20 OCT 1 NOT EVEN IF THERE'S A RESTART.
21
22 CA Z
23 TS DISPDEX # KILL CLOKTASK
24 TC P40/RET
25
26
27
28
29
30
31
32
33
34 ASTNRETN TC PHASCHNG
35 OCT 04024
36
37 CAF ZERO # STOP DISPLAYING BUT KEEP RUNNING
38 TS DISPDEX
39 CAF PRI013
40 TC FINDVAC
41 EBANK= STARIND
42 2CADR ASTNRET
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```



INHINT # SET UP THE DAP FOR COASTING FLIGHT.

TC IBNKCALL

CADR ALLCOAST

TC NULLCLOK

TC PHASCHNG

# 4.13 RESTART FOR POSTBURN

OCT 00134

TCF ENDOFJOB

GOCUTOFF

CAF

PRI017

# (3)

TC

FINDVAC

EBANK=

TGO

2CADR

CUTOFF

TC

DOWNFLAG

ADRES

FLUNDISP

INHINT

# SET UP THE DAP FOR COASTING FLIGHT.

TC

IBNKCALL

CADR

ALLCOAST

TC

NULLCLOK

TC

PHASCHNG

OCT

07024

OCT

17000

EBANK=

TGO

2CADR

CUTOFF

TCF

ENDOFJOB

IGNITE

CS

FLAGWRD7

# (2)

MASK

IGNFLBIT

CCS

A

TCF

IGNITE1

CAF

BIT1

INHINT

TC

TWIDDLE

ADRES

IGNITION

CAF

OCT23

# IMMEDIATE RESTART AT IGNITION

TS

L

COM

DXCH

-PHASE4

IGNITE1

CS

CNTDNDEX

# RESTORE OLD DISPLAY.

TS

DISPDEX

TCF

ENDOFJOB

# \*\*\*\*\*

|        |     |       |                                         |
|--------|-----|-------|-----------------------------------------|
| P40ALM | TC  | ALARM | # PROGRAM SELECTION NOT CONSISTENT WITH |
|        | OCT | 1706  | # VEHICLE CONFIGURATION                 |

|          |      |          |        |
|----------|------|----------|--------|
| REP40ALM | CAF  | V05N09   | # (14) |
|          | TC   | BANKCALL |        |
|          | CADR | GOFLASH  |        |

|     |          |           |                 |
|-----|----------|-----------|-----------------|
| TCF | GOTOP00H | # V34E    | TERMINATE       |
| TCF | +2       | # PROCEED | CHECK FOR P42   |
| TCF | REP40ALM | # V32E    | REDISPLAY ALARM |

|       |       |                                       |
|-------|-------|---------------------------------------|
| INDEX | WHICH | # FOR P42, ALLOW CREW TO PROCEED EVEN |
| TCF   | 14    | # THOUGH VEHICLE IS UNSTAGED.         |

# \*\*\*\*\*

|        |       |
|--------|-------|
| BANK   | 31    |
| SETLOC | P40S2 |
| BANK   |       |

COUNT\* \$\$/P40

|         |    |          |                                          |
|---------|----|----------|------------------------------------------|
| P40AUTO | TC | MAKECADR | # HELLO THERE.                           |
|         | TS | TEMPR60  | # FOR GENERALIZED RETURN TO OTHER BANKS. |

|        |      |          |                                         |
|--------|------|----------|-----------------------------------------|
| P40A/P | TC   | BANKCALL | # SUBROUTINE TO CHECK PGNC'S CONTROL    |
|        | CADR | G+N,AUTO | # AND AUTO STABILIZATION MODES          |
|        | CCS  | A        | # +0 INDICATES IN PGNC'S, IN AUTO       |
|        | TCF  | TURNITON | # + INDICATES NOT IN PGNC'S AND/OR AUTO |
|        | CAF  | APSFLBIT | # ARE WE ON THE DESCENT STAGE?          |
|        | MASK | FLGWRD10 |                                         |

|     |        |                                     |
|-----|--------|-------------------------------------|
| CCS | A      |                                     |
| TCF | GOBACK | # RETURN                            |
| CAF | BIT5   | # YES, CHECK FOR AUTO-THROTTLE MODE |

|        |        |
|--------|--------|
| EXTEND |        |
| RAND   | CHAN30 |
| EXTEND |        |

|          |     |          |                                         |
|----------|-----|----------|-----------------------------------------|
| TURNITON | BZF | GOBACK   | # IN AUTO-THROTTLE MODE -- RETURN       |
|          | CAF | P40A/PMD | # DISPLAYS V50N25 R1=203 PLEASE PERFORM |
|          | TC  | BANKCALL | # CHECKLIST 203 TURN ON PGNC'S ETC.     |

|      |          |                  |
|------|----------|------------------|
| CADR | GOPERF1  |                  |
| TCF  | GOTOP00H | # V34E TERMINATE |
| TCF  | P40A/P   | # RECYCLE        |

|        |    |          |                             |
|--------|----|----------|-----------------------------|
| GOBACK | CA | TEMPR60  |                             |
|        | TC | BANKJUMP | # GOODBYE. COME AGAIN SOON. |

|          |     |       |
|----------|-----|-------|
| P40A/PMD | OCT | 00203 |
|----------|-----|-------|

|    |          |                                    |             |    |
|----|----------|------------------------------------|-------------|----|
| 1  |          |                                    |             | 1  |
| 2  |          | BANK                               | 36          | 2  |
| 3  |          | SETLOC                             | P40S        | 3  |
| 4  |          | BANK                               |             | 4  |
| 5  |          |                                    |             | 5  |
| 6  |          | COUNT*                             | \$\$/P40    | 6  |
| 7  |          |                                    |             | 7  |
| 8  | #        | *****                              |             | 8  |
| 9  | #        | CONSTANTS FOR THE IGNITION ROUTINE |             | 9  |
| 10 | #        | *****                              |             | 10 |
| 11 |          |                                    |             | 11 |
| 12 | SERVCADR | =                                  | P63TABLE +7 | 12 |
| 13 |          |                                    |             | 13 |
| 14 | P40ADRES | ADRES                              | P40TABLE    | 14 |
| 15 |          |                                    |             | 15 |
| 16 | P41ADRES | ADRES                              | P41TABLE -5 | 16 |
| 17 |          |                                    |             | 17 |
| 18 | P42ADRES | ADRES                              | P42TABLE    | 18 |
| 19 |          |                                    |             | 19 |
| 20 |          | EBANK=                             | DVCNTR      | 20 |
| 21 | DSP2CADR | 2CADR                              | P63DISPS -2 | 21 |
| 22 |          |                                    |             | 22 |
| 23 |          | EBANK=                             | DVCNTR      | 23 |
| 24 | ATMAGADR | 2CADR                              | ATMAG       | 24 |
| 25 |          |                                    |             | 25 |
| 26 | ?        | =                                  | GOTOP00H    | 26 |
| 27 |          |                                    |             | 27 |
| 28 | D29.9SEC | 2DEC                               | 2990        | 28 |
| 29 |          |                                    |             | 29 |
| 30 | S24.9SEC | DEC                                | 2490        | 30 |
| 31 |          |                                    |             | 31 |
| 32 | 4.9SEC   | DEC                                | 490         | 32 |
| 33 |          |                                    |             | 33 |
| 34 | OCT20    | =                                  | BIT5        | 34 |
| 35 |          |                                    |             | 35 |
| 36 | V06N61   | VN                                 | 0661        | 36 |
| 37 |          |                                    |             | 37 |
| 38 |          |                                    |             | 38 |
| 39 |          |                                    |             | 39 |
| 40 |          |                                    |             | 40 |
| 41 |          |                                    |             | 41 |
| 42 |          |                                    |             | 42 |
| 43 |          |                                    |             | 43 |
| 44 |          |                                    |             | 44 |
| 45 |          |                                    |             | 45 |
| 46 |          |                                    |             | 46 |
| 47 |          |                                    |             | 47 |
| 48 |          |                                    |             | 48 |
| 49 |          |                                    |             | 49 |
| 50 |          |                                    |             | 50 |
| 51 |          |                                    |             | 51 |
| 52 |          |                                    |             | 52 |
| 53 |          |                                    |             | 53 |
| 54 |          |                                    |             | 54 |
| 55 |          |                                    |             | 55 |
| 56 |          |                                    |             | 56 |
| 57 |          |                                    |             | 57 |
| 58 |          |                                    |             | 58 |
| 59 |          |                                    |             | 59 |
| 60 |          |                                    |             | 60 |

# KILLTASK

# MOD NO: NEW PROGRAM

# MOD BY: COVELLI

#

# FUNCTIONAL DESCRIPTION:

#

# KILLTASK IS USED TO REMOVE A TASK FROM THE WAITLIST BY SUBSTITUTING A NULL TASK CALLED 'NULLTASK' (OF COURSE), WHICH MERELY DOES A TC TASKOVER. IF THE SAME TASK IS SCHEDULED MORE THAN ONCE, ONLY THE ONE WHICH WILL OCCUR

# FIRST IS REMOVED. IF THE TASK IS NOT SCHEDULED, KILLTASK TAKES NO ACTION AND RETURNS WITH NO ALARM. KILLTASK LEAVES INTERRUPTS INHIBITED SO CALLER MUST RELINT

#

# CALLING SEQUENCE

# L TC KILLTASK # IN FIXED-FIXED

# L+1 CADR ???????? # CADR (NOT 2CADR) OF TASK TO BE REMOVED.

# L+2 (RELINT) # RETURN

#

# EXIT MODE: AT L+2 OF CALLING SEQUENCE.

#

# ERASABLE INITIALIZATION: NONE.

#

# OUTPUT: 2CADR OF NULLTASK IN LST2

#

# DEBRIS: ITEMP1 - ITEMP4, A, L, Q.

EBANK= LST2

BLOCK 3

# KILLTASK MUST BE IN FIXED-FIXED.

SETLOC FFTAG6

BANK

COUNT\* \$\$/KILL

KILLTASK

CA KILLBB

INHINT

LXCH A

INDEX Q

CA 0

# GET CADR.

LXCH BBANK

TCF

KILLTSK2

# CONTINUE IN SWITCHED FIXED.

EBANK= LST2

KILLBB

BBCON KILLTSK2

BANK 27

SETLOC P40S1

BANK

COUNT\* \$\$/KILL

KILLTSK2

LXCH ITEMP2

# SAVE CALLER'S BBANK

```
1
2 INCR Q
3 EXTEND
4 QXCH ITEMP1 # RETURN 2ADR IN ITEMP1,ITEMP2
5
6 TS ITEMP3 # CADR IS IN A
7 MASK LOW10
8 AD BIT11
9 TS ITEMP4 # GENADR OF TASK
10
11 CS LOW10
12 MASK ITEMP3
13 TS ITEMP3 # FBANK OF TASK
14
15 ZL
16 ADRSCAN INDEX L
17 CS LST2
18 AD ITEMP4 # COMPARE GENADRS
19
20 LETITLIV EXTEND
21 BZF TSTFBANK # IF THEY MATCH, COMPARE FBANKS
22 CS LSTLIM
23 AD L
24 EXTEND
25 BZF DEAD # ARE WE DONE?
26 BZF DEAD # YES -- DONE, SO RETURN
27 INCR L
28 INCR L
29 TCF ADRSCAN # CONTINUE LOOP.
30
31 DEAD DXCH ITEMP1
32 DTCB
33
34 TSTFBANK CS LOW10
35 INDEX L
36 MASK LST2 +1 # COMPARE FBANKS ONLY.
37 EXTEND
38 SU ITEMP3
39 EXTEND
40 BZF KILLDEAD # MATCH -- KILL IT.
41 TCF LETITLIV # NO MATCH -- CONTINUE.
42
43 KILLDEAD CA TCTSKOVR
44 INDEX L
45 TS LST2 # REMOVE TASK BY INSERTING TASKOVER
46 TCF DEAD
47
48 LSTLIM EQUALS BIT5 # DEC 16
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1 # PROGRAM DESCRIPTION: P40BOTH DECEMBER 22, 1966
2 # MOD 03 BY PETER ADLER MARCH 3, 1967
3 # CALLED VIA JOB FROM V37E
4 #
5 # FUNCTIONAL DESCRIPTION
6 #
7 # 1) TO COMPUTE A PREFERRED IMU ORIENTATION AND A PREFERRED VEHICLE ATTITUDE FOR A LM DPS
8 # THRUSTING MANEUVER.
9 # (THERE IS NO ITEM #2 IN THE ORIGINAL PROGRAM LISTING --- RSB 2009.)
10 # 3) TO DO THE VEHICLE MANEUVER TO THE THRUSTING ATTITUDE.
11 # 4) TO CONTROL THE PGNC'S DURING COUNTDOWN, IGNITION, THRUSTING, AND THRUST TERMINATION OF A
12 # PGNC'S CONTROLLED DPS MANEUVER.
13 # 5) IN POSTBURN --- ZERO RENDEZVOUS COUNTER, MAINTAIN VG CALCULATIONS FOR POSSIBLE RCS MANEUVER,
14 # SET MAXIMUM DEADBAND IN DAP, RESET STEERLAW CSTEER TO ZERO.
15 #
16 # NOTE: P42, WHICH IS IN THIS LOG SECTION, DOES THE SAME FOR AN APS BURN, AND P41 DOES 1-3 FOR
17 # RCS PLUS DISPLAYS PARAMETERS FOR MANUAL CONTROL.
18 #
19 # SUBROUTINES USED
20 #
21 # R02 IMU STATUS CHECK
22 # S40.1 COMPUTATION OF THRUST DIRECTION
23 # S40.13 LENGTH OF BURN
24 # S40.2,3 PREFERRED IMU ORIENTATION
25 # S40.8 X PRODUCT STEERING
26 # S40.9 LAMBERT VTOGAIN
27 # R60LEM ATTITUDE MANEUVER
28 # LEMPREC EXTRAPOLATE STATE VECTOR
29 # PREREAD AVERAGE G, SERVICER
30 # ALLCOAST DAP COASTING INITIALIZATION
31 # CLOKTASK ERGO CLOCKJOB -- COUNT DOWN
32 # PHASCHANG, INTPRET, FLAGUP, FLAGDOWN, WAITLIST, LONGCALL, GOFLASH, GOFLASHR, GOPERF1, ALARM,
33 # PRIOLARM, GOTOPOOH, ENDOFJOB, BANKCALL, SETMAXDB, SETMINDB, CHECKMM, FLATOUT, OUTFLAT,
34 # KILLTASK, SGNAGREE, TPAGREE, ETC.
35 #
36 # RESTARTS VIA GROUP 4
37 #
38 # DISPLAYS
39 #
40 # V50N25 203 A/P TO PGNC'S, AUTO-THROTTLE MODE, AUTO ATTITUDE CONTROL
41 # V06N40 TTI, VG, DELTAVM (DISPLAYED ONCE/SECOND BY CLOKTASK)
42 # V50N99 PLEASE PERFORM ENGINE ON ENABLE
43 # V06N40 TG (TIME TO GO TO CUTOFF), VG, DELTAVM -- ONCE/SECOND
44 # V16N40 FINAL VALUES OF TG, VG, DELTAVM
45 # V16N85 COMP OF VG (BODY AXES) FOR POSS. RCS MANUAL MANEUVER
46 # V05N09 POSSIBLE ALARMS
47 # V50N07 PLEASE SELECT P00
48 #
49 #
50 #
51 #
52 #
53 #
54 #
55 #
56 #
57 #
58 #
59 #
60 #
```

# VIA R30

# V06N44 HAP0, PERI, TFF  
# V06N35 TIME TO PERIGEE, HMS

# ALARM OR ABORT EXIT MODES

# PROGRAM ALARM, FLASHING DISPLAY OF ALARM CODE 1706 IF P40 SELECTED WITH DESCENT UNIT STAGED.  
# V34E (TERMINATE) IS THE ONLY RESPONSE ACCEPTED. TC GOTOP00H.# PROGRAM ALARM, FLASH CODE 1703: TIG LESS THAN 45 SECS AWAY. V34E= GOTOP00H OR V33E= SLIP  
# TIG BY 45 SECS.

# ERASABLE INITIALIZATION

# DEBRIS

# OUTPUT

# SEE SUBROUTINES E.G.: S40.1, S40.2,3, S40.13, S40.8, S40.9, TRIMGIMB  
# XDELVFLG = 1 FOR EXT DELV COMPUTATION  
# = 0 FOR AIMPT (LAMBERT COMPCOUNT\* \$\$/P40  
EBANK= WHICHBANK 36  
SETLOC P40S  
BANKP40LM TC PHASCHNG  
OCT 04024CAF P40ADRES # INITIALIZATION FOR BURNBABY  
TS WHICHCA FLGWRD10  
MASK APSFLBITCCS A  
TCF P40ALM  
TC BANKCALL # GO DO IMU STATUS CHECK ROUTINE.  
CADR R02BOTH

CS DAPBOOLS # INITIALIZE DVMON

MASK CSMDOCKD  
CCS A  
CAF THRESH1  
AD THRESH3  
TS DVTHRUSH  
CAF FOUR  
TS DVCNTR

[illegible]



P40PHS1

TC PHASCHNG  
OCT 00014  
TCF ENDOFJOB

TIGNOW

INHINT  
TC IBNKCALL  
CADR ZATTEROR  
TC IBNKCALL  
CADR SETMINDB  
RELINT  
CAF V16N85B  
TC BANKCALL  
CADR REFLASHR  
TC TERM40  
TCF TERM40  
TC -5

TERM40

TCF P40PHS1  
EXTEND  
DCA SERVCADR  
DXCH AVEGEXIT  
CAF ZERO  
TS TRKMKCNT # ZERO RENDZVS CNTERS  
CA Z  
TS DISPDEXINHINT  
TC IBNKCALL  
CADR RESTORDBRELINT  
TC GOTOPOOH

P41LM

EBANK= WHICH  
COUNT\* \$\$/P41  
CAF P41ADRES # INITIALIZATION FOR BURNBABY  
TS WHICHTC BANKCALL  
CADR R02BOTHTC INTPRET # BOTH LM  
BON DLOAD # IF NJETSFLAG IS SET, LOAD Z JET F  
NJETSFLG  
P41FJET1  
FRCS4 # IF NJETSFLG IS CLEAR, LOAD 4 JET F

P41FJET

STCALL F

P41FJET1

DLOAD P41IN

|    |           |        |                                           |    |
|----|-----------|--------|-------------------------------------------|----|
| 1  | # P41NORM |        |                                           | 1  |
| 2  |           |        | FRCS2                                     | 2  |
| 3  |           | STORE  | F                                         | 3  |
| 4  |           |        |                                           | 4  |
| 5  | P41IN     | CALL   |                                           | 5  |
| 6  |           |        | S40.1                                     | 6  |
| 7  |           |        | # BOTH                                    | 7  |
| 8  | P41NORM   | CALL   |                                           | 8  |
| 9  |           |        | S40.2,3                                   | 9  |
| 10 |           | EXIT   |                                           | 10 |
| 11 |           |        | # CALCULATE PREFERRED IMU ORIENTATION AND | 11 |
| 12 |           |        | # SET PFRATFLG.                           | 12 |
| 13 |           | INHINT |                                           | 13 |
| 14 |           | TC     | IBNKCALL                                  | 14 |
| 15 |           | CADR   | ZATTEROR                                  | 15 |
| 16 |           |        | # ZERO ATTITUDE ERRORS                    | 16 |
| 17 |           | TC     | IBNKCALL                                  | 17 |
| 18 |           | CADR   | SETMINDB                                  | 18 |
| 19 |           |        | # SET 0.3 DEGREE DEADBAND                 | 19 |
| 20 |           | TC     | P40SXT4                                   | 20 |
| 21 |           |        |                                           | 21 |
| 22 |           | TC     | INTPRET                                   | 22 |
| 23 |           | VLOAD  | CALL                                      | 23 |
| 24 |           |        | # TRANSFORM VELOCITY-TO-BE-GAINED AT TIG  | 24 |
| 25 |           |        | # FROM REFERENCE COORDINATES TO LM BODY-  | 25 |
| 26 |           |        | VGTIG                                     | 26 |
| 27 |           |        | S41.1                                     | 27 |
| 28 |           |        | # AXIS COORDINATES FOR V16N85 DISPLAY.    | 28 |
| 29 |           | STORE  | VGBODY                                    | 29 |
| 30 |           | EXIT   |                                           | 30 |
| 31 |           |        | # (SCALED AT 2 (+7) METERS/CENTISECOND)   | 31 |
| 32 |           |        |                                           | 32 |
| 33 |           | CAF    | V16N85B                                   | 33 |
| 34 |           | TC     | BANKCALL                                  | 34 |
| 35 |           | CADR   | GODSPRET                                  | 35 |
| 36 |           |        |                                           | 36 |
| 37 |           | CAF    | PRI05                                     | 37 |
| 38 |           | TS     | DISPDEX                                   | 38 |
| 39 |           |        | # FOR SAFETY ONLY                         | 39 |
| 40 |           | TC     | FINDVAC                                   | 40 |
| 41 |           | EBANK= | VGPREV                                    | 41 |
| 42 |           | 2CADR  | DYNMDISP                                  | 42 |
| 43 |           |        |                                           | 43 |
| 44 |           | TC     | 2PHSCHNG                                  | 44 |
| 45 |           | OCT    | 00076                                     | 45 |
| 46 |           |        | # GROUP 6 RESTARTS AT RED06.7             | 46 |
| 47 |           | OCT    | 04024                                     | 47 |
| 48 |           |        | # GROUP 4 RESTARTS HERE                   | 48 |
| 49 | #         |        | *****                                     | 49 |
| 50 |           |        |                                           | 50 |
| 51 |           | TCF    | B*RNB*B*                                  | 51 |
| 52 |           |        |                                           | 52 |
| 53 | #         |        | *****                                     | 53 |
| 54 |           |        |                                           | 54 |
| 55 | BLNKWAIT  | CAF    | 1SEC                                      | 55 |
| 56 |           | TC     | BANKCALL                                  | 56 |
| 57 |           | CADR   | DELAYJOB                                  | 57 |
| 58 |           |        |                                           | 58 |
| 59 | RED06.7   | CA     | DISPDEX                                   | 59 |
| 60 |           | AD     | TWO                                       | 60 |
| 61 |           |        | # ON A RESTART, DO NOT PUT UP DISPLAY IF  | 61 |
| 62 |           |        | # BLANKING (BETWEEN TIG-35 AND TIG-30)    | 62 |
| 63 | #         |        | *****                                     | 63 |
| 64 |           |        |                                           | 64 |
| 65 |           |        |                                           | 65 |
| 66 |           |        |                                           | 66 |
| 67 |           |        |                                           | 67 |
| 68 |           |        |                                           | 68 |
| 69 |           |        |                                           | 69 |
| 70 |           |        |                                           | 70 |
| 71 |           |        |                                           | 71 |
| 72 |           |        |                                           | 72 |
| 73 |           |        |                                           | 73 |
| 74 |           |        |                                           | 74 |
| 75 |           |        |                                           | 75 |
| 76 |           |        |                                           | 76 |
| 77 |           |        |                                           | 77 |
| 78 |           |        |                                           | 78 |
| 79 |           |        |                                           | 79 |
| 80 |           |        |                                           | 80 |

|    |             |        |          |                                         |          |
|----|-------------|--------|----------|-----------------------------------------|----------|
| 1  | # P40 P41   |        |          |                                         | PAGE 131 |
| 2  |             |        |          |                                         |          |
| 3  | EXTEND      |        |          |                                         |          |
| 4  |             | BZF    | BLNKWAIT |                                         |          |
| 5  |             |        |          |                                         |          |
| 6  |             | CAF    | V16N85B  |                                         |          |
| 7  | TC BANKCALL |        |          |                                         |          |
| 8  |             | CADR   | GODSPRET |                                         |          |
| 9  |             |        |          |                                         |          |
| 10 |             | CAF    | PRI05    |                                         |          |
| 11 |             | TC     | PRI0CHNG |                                         |          |
| 12 |             |        |          |                                         |          |
| 13 | DYNMDISP    | CA     | DISPDEX  | # A NON-POSITIVE DISPDEX INDICATES PAST |          |
| 14 |             | EXTEND |          | # TIG-35, SO SERVICER WILL BE DOING THE |          |
| 15 |             | BZMF   | ENDOFJOB | # UPDATING OF NOUN 85. STOP DYNMDISP.   |          |
| 16 |             | TC     | INTPRET  |                                         |          |
| 17 |             | VLOAD  | CALL     |                                         |          |
| 18 |             |        | VGPREV   |                                         |          |
| 19 | S41.1       |        |          |                                         |          |
| 20 |             | STORE  | VGBODY   |                                         |          |
| 21 |             | EXIT   |          |                                         |          |
| 22 |             | CAF    | 1SEC     |                                         |          |
| 23 |             | TC     | BANKCALL |                                         |          |
| 24 |             | CADR   | DELAYJOB |                                         |          |
| 25 |             | TCF    | DYNMDISP |                                         |          |
| 26 |             |        |          |                                         |          |
| 27 | CALCN85     | TC     | INTPRET  |                                         |          |
| 28 |             | CALL   |          |                                         |          |
| 29 |             |        | UPDATEVG |                                         |          |
| 30 |             | VLOAD  | CALL     |                                         |          |
| 31 | VGPREV      |        |          |                                         |          |
| 32 | S41.1       |        |          |                                         |          |
| 33 |             | STORE  | VGBODY   |                                         |          |
| 34 |             | EXIT   |          |                                         |          |
| 35 |             | TC     | POSTJUMP |                                         |          |
| 36 |             | CADR   | SERVEXIT |                                         |          |
| 37 |             |        |          |                                         |          |
| 38 |             | COUNT* | \$\$/P42 |                                         |          |
| 39 |             | EBANK= | WHICH    |                                         |          |
| 40 |             |        |          |                                         |          |
| 41 | P42LM       | TC     | PHASCHNG |                                         |          |
| 42 |             | OCT    | 04024    |                                         |          |
| 43 |             |        |          |                                         |          |
| 44 |             | CAF    | P42ADRES | # INITIALIZATION FOR BURNBABY.          |          |
| 45 |             | TS     | WHICH    |                                         |          |
| 46 |             |        |          |                                         |          |
| 47 |             | CS     | FLGWRD10 |                                         |          |
| 48 |             | MASK   | APSFLBIT |                                         |          |
| 49 | A           |        |          |                                         |          |
| 50 |             | CCS    | P40ALM   |                                         |          |
| 51 | P42STAGE    | TC     | BANKCALL |                                         |          |
| 52 |             |        |          |                                         |          |
| 53 |             |        |          |                                         |          |
| 54 |             |        |          |                                         |          |
| 55 |             |        |          |                                         |          |
| 56 |             |        |          |                                         |          |
| 57 |             |        |          |                                         |          |
| 58 |             |        |          |                                         |          |
| 59 |             |        |          |                                         |          |
| 60 |             |        |          |                                         |          |

1412THE

1

[illegible]

|    |          |        |          |                                            |
|----|----------|--------|----------|--------------------------------------------|
| 1  |          |        |          |                                            |
| 2  |          | TC     | NOULLAGE | # TURN OFF ULLAGE                          |
| 3  |          |        |          |                                            |
| 4  |          | TC     | TASKOVER |                                            |
| 5  |          |        |          |                                            |
| 6  | ENGOFTSK | TC     | IBNKCALL | # THIS CODING ALLOWS ENGINOFF ET AL TO BE  |
| 7  |          | CADR   | ENGINOFF | # USED BOTH BY WAITLIST AND BY TC IBNKCALL |
| 8  |          | TC     | TASKOVER |                                            |
| 9  |          |        |          |                                            |
| 10 | ENGINOFF | CAF    | PRI012   | # MUST BE LOWER PRI0 THAN CLOCKJOB         |
| 11 |          | TC     | FINDVAC  |                                            |
| 12 |          | EBANK= | TRKMKCNT |                                            |
| 13 |          | 2CADR  | POSTBURN |                                            |
| 14 |          |        |          |                                            |
| 15 | ENGINOF2 | CAF    | BIT1     |                                            |
| 16 |          | TC     | WAITLIST |                                            |
| 17 |          | EBANK= | OMEGAQ   |                                            |
| 18 |          | 2CADR  | COASTSET |                                            |
| 19 |          |        |          |                                            |
| 20 | ENGINOF1 | CS     | FLAGWRD7 | # SET THE IDLE BIT.                        |
| 21 |          | MASK   | IDLEFBIT |                                            |
| 22 |          | ADS    | FLAGWRD7 |                                            |
| 23 |          |        |          |                                            |
| 24 |          | TC     | NOULLAGE |                                            |
| 25 |          |        |          |                                            |
| 26 | ENGINOF4 | EXTEND |          |                                            |
| 27 |          | DCA    | TIME2    |                                            |
| 28 |          | DXCH   | TEVENT   |                                            |
| 29 |          |        |          |                                            |
| 30 | ENGINOF3 | CS     | ENGONBIT | # INSURE ENGONFLG IS CLEAR.                |
| 31 |          | MASK   | FLAGWRD5 |                                            |
| 32 |          | TS     | FLAGWRD5 |                                            |
| 33 |          | CS     | PRI030   | # ENGINOF3 IS USED AS A PRE-ENGINE ARM     |
| 34 |          | EXTEND |          | # SUBROUTINE.                              |
| 35 |          | RAND   | DSALMOUT |                                            |
| 36 |          | AD     | PRI020   | # TURN OFF THE ENGINE -- DPS OR APS        |
| 37 |          | EXTEND |          |                                            |
| 38 |          | WRITE  | DSALMOUT |                                            |
| 39 |          |        |          |                                            |
| 40 |          | CS     | DAPBOOLS | # TURN OFF TRIM GIMBAL                     |
| 41 |          | MASK   | USEQRJTS |                                            |
| 42 |          | ADS    | DAPBOOLS |                                            |
| 43 |          |        |          |                                            |
| 44 |          | CS     | HIRTHROT | # ZERO AUTO-THROTTLE WHENEVER THE ENGINE   |
| 45 |          | TS     | THRUST   | # IS TURNED OFF.                           |
| 46 |          | CAF    | BIT4     | # THE HARDWARE DOES SO ONLY WHEN THE       |
| 47 |          | EXTEND |          | # ENGINE IS DISARMED.                      |
| 48 |          | WOR    | CHAN14   |                                            |
| 49 |          |        |          |                                            |
| 50 |          | TC     | ISWRETRN |                                            |
| 51 |          |        |          |                                            |
| 52 |          |        |          |                                            |
| 53 |          |        |          |                                            |
| 54 |          |        |          |                                            |
| 55 |          |        |          |                                            |
| 56 |          |        |          |                                            |
| 57 |          |        |          |                                            |
| 58 |          |        |          |                                            |
| 59 |          |        |          |                                            |
| 60 |          |        |          |                                            |

|    |          |        |          |                                   |    |
|----|----------|--------|----------|-----------------------------------|----|
| 1  |          |        |          |                                   | 1  |
| 2  | COASTSET | TC     | IBNKCALL | # DO DAP COASTING INITIALIZATION  | 2  |
| 3  |          | CADR   | ALLCOAST |                                   | 3  |
| 4  |          | TC     | TASKOVER |                                   | 4  |
| 5  |          |        |          |                                   | 5  |
| 6  |          | EBANK= | OMEGAQ   |                                   | 6  |
| 7  | UPDATEVG | STQ    | CALL     |                                   | 7  |
| 8  |          |        | QTEMP1   |                                   | 8  |
| 9  |          |        | S40.8    | # X-PRODUCT STEERING              | 9  |
| 10 |          | BON    | BON      |                                   | 10 |
| 11 |          |        | XDELVFLG |                                   | 11 |
| 12 |          |        | QTEMP1   |                                   | 12 |
| 13 |          |        | NORMSW   |                                   | 13 |
| 14 |          |        | 180SETUP |                                   | 14 |
| 15 |          | DLOAD  | DSU      |                                   | 15 |
| 16 |          |        | PIPTIME  |                                   | 16 |
| 17 |          |        | TIGSAVE  |                                   | 17 |
| 18 |          | DSU    | BMN      |                                   | 18 |
| 19 |          |        | TNEWA    |                                   | 19 |
| 20 |          |        | GETRANS  |                                   | 20 |
| 21 |          | DLOAD  | DAD      |                                   | 21 |
| 22 |          |        | TIGSAVE  |                                   | 22 |
| 23 |          |        | TNEWA    |                                   | 23 |
| 24 |          | STORE  | TIGSAVEP |                                   | 24 |
| 25 | 180SETUP | EXIT   |          |                                   | 25 |
| 26 |          | CCS    | PHASE2   |                                   | 26 |
| 27 |          | TCF    | NO.9     |                                   | 27 |
| 28 |          | CAF    | PRI010   |                                   | 28 |
| 29 |          | INHINT |          |                                   | 29 |
| 30 |          | TC     | FINDVAC  |                                   | 30 |
| 31 |          | EBANK= | VG       |                                   | 31 |
| 32 |          | 2CADR  | S40.9    | # LAMBERT VTOGAIN                 | 32 |
| 33 |          |        |          |                                   | 33 |
| 34 |          | TC     | 2PHSCHNG |                                   | 34 |
| 35 |          | OCT    | 00172    | # 2.17SPOT FOR S40.9              | 35 |
| 36 |          | OCT    | 10035    | # HERE AND REREADAC AFTER RESTART | 36 |
| 37 |          |        |          |                                   | 37 |
| 38 | ENDSTEER | TC     | INTPRET  |                                   | 38 |
| 39 |          | DLOAD  |          |                                   | 39 |
| 40 |          |        | TIGSAVEP |                                   | 40 |
| 41 |          | STOVL  | TIGSAVE  |                                   | 41 |
| 42 |          |        | RN       |                                   | 42 |
| 43 |          | STOVL  | RINIT    |                                   | 43 |
| 44 |          |        | VN       |                                   | 44 |
| 45 |          | STORE  | VINIT    |                                   | 45 |
| 46 | GETRANS  | DLOAD  | DSU      |                                   | 46 |
| 47 |          |        | TPASS4   |                                   | 47 |
| 48 |          |        | PIPTIME  |                                   | 48 |
| 49 |          | STCALL | DELLT4   |                                   | 49 |
| 50 |          |        | QTEMP1   |                                   | 50 |
| 51 |          |        |          |                                   | 51 |
| 52 |          |        |          |                                   | 52 |
| 53 |          |        |          |                                   | 53 |
| 54 |          |        |          |                                   | 54 |
| 55 |          |        |          |                                   | 55 |
| 56 |          |        |          |                                   | 56 |
| 57 |          |        |          |                                   | 57 |
| 58 |          |        |          |                                   | 58 |
| 59 |          |        |          |                                   | 59 |
| 60 |          |        |          |                                   | 60 |

NO.9

TC  
GOTO

INTPRET

STEERING

TC

QTEMP1  
INTPRET

CALL

UPDATEVG

EXIT

NSTEER

EBANK=  
INHINT

DVCNTR

CA

EBANK7

TS

EBANK

CS

FLAGWRD2

# CHECK IMPULSE SWITCH. IT IS SET EITHER

MASK

IMPULBIT

# BY S40.13 IF TBURN&lt;6 SECS OR BY S40.8 IF

CCS

A

# STEERING IS ALMOST DONE.

TCF

+5

# IMPULSW = 0 EXIT

CS

FLAGWRD7

# IMPULSW = 1 WHY? CHECK IDLEFLAG

MASK

IDLEFBIT

# (IDLEFLAG = 0 --&gt; DVMON ON)

CCS

A

TCF

+3

# DVMON ON --&gt; THRUSTING --&gt; IMPULSW VIA S40.8

TC

POSTJUMP

# DVMON OFF --&gt; IMPULSW ON VIA S40.13 --&gt; EXIT

CADR

SERVEXIT

TC

IBNKCALL

CADR

STOPRATE

TC

DOWNFLAG

# TURN OFF IMPULSW

ADRES

IMPULSW

TC

UPFLAG

ADRES

IDLEFLAG

# TURN OFF DVMON

INHINT

EXTEND

DCA

TIG

DXCH

MPAC

EXTEND

DCS

TIME2

DAS

MPAC

TC

TPAGREE

CAE

MPAC +1

TC

GETDT

TC

TWIDDLE

ADRES

ENGOFTSK

TC

2PHSCHNG

OCT

40114

# ENGOFTSK (ENGINEOFF)

OCT

00035

# SERVICER -- REREADAC



|    |         |        |             |                                     |    |
|----|---------|--------|-------------|-------------------------------------|----|
| 1  |         |        |             |                                     | 1  |
| 2  |         | TCF    | ENDOFJOB    |                                     | 2  |
| 3  |         |        |             |                                     | 3  |
| 4  | GETDT   | CCS    | A           |                                     | 4  |
| 5  |         | TCF    | +3          |                                     | 5  |
| 6  |         | TCF    | +2          |                                     | 6  |
| 7  |         | CAF    | ZERO        |                                     | 7  |
| 8  |         | AD     | ONE         |                                     | 8  |
| 9  |         | XCH    | L           |                                     | 9  |
| 10 |         | CAF    | ZERO        |                                     | 10 |
| 11 |         | DXCH   | TGO         |                                     | 11 |
| 12 |         | CA     | TGO         | +1                                  | 12 |
| 13 |         | TC     | Q           |                                     | 13 |
| 14 |         |        |             |                                     | 14 |
| 15 | # ***** |        |             |                                     | 15 |
| 16 |         |        |             |                                     | 16 |
| 17 | SEC15DP | OCT    | 00000       | # DON'T SEPARATE                    | 17 |
| 18 | SEC15   | DEC    | 1500        | # DON'T SEPARATE                    | 18 |
| 19 | SEC30DP | 2DEC   | 3000        |                                     | 19 |
| 20 |         |        |             |                                     | 20 |
| 21 | SEC45DP | OCT    | 00000       | # DON'T MOVE FROM JUST BEFORE SEC45 | 21 |
| 22 | SEC45   | DEC    | 4500        |                                     | 22 |
| 23 | 5SECDP  | OCT    | 00000       | # DON'T MOVE FROM JUST BEFORE 5SEC  | 23 |
| 24 | 5SEC    | DEC    | 500         |                                     | 24 |
| 25 | 26SECS  | DEC    | 2600        |                                     | 25 |
| 26 | V16N40  | VN     | 1640        |                                     | 26 |
| 27 | V16N85B | VN     | 1685        |                                     | 27 |
| 28 | V1683   | VN     | 1683        |                                     | 28 |
| 29 | SEC01   | =      | 1SEC        |                                     | 29 |
| 30 | ACADN85 | =      | P41TABLE +2 |                                     | 30 |
| 31 |         |        |             |                                     | 31 |
| 32 |         | EBANK= | DELVIMU     |                                     | 32 |
| 33 | ACADN83 | 2CADR  | CALCN83     |                                     | 33 |
| 34 |         |        |             |                                     | 34 |
| 35 | # ***** |        |             |                                     | 35 |
| 36 |         |        |             |                                     | 36 |
| 37 |         |        |             |                                     | 37 |
| 38 |         |        |             |                                     | 38 |
| 39 |         |        |             |                                     | 39 |
| 40 |         |        |             |                                     | 40 |
| 41 |         |        |             |                                     | 41 |
| 42 |         |        |             |                                     | 42 |
| 43 |         |        |             |                                     | 43 |
| 44 |         |        |             |                                     | 44 |
| 45 |         |        |             |                                     | 45 |
| 46 |         |        |             |                                     | 46 |
| 47 |         |        |             |                                     | 47 |
| 48 |         |        |             |                                     | 48 |
| 49 |         |        |             |                                     | 49 |
| 50 |         |        |             |                                     | 50 |
| 51 |         |        |             |                                     | 51 |
| 52 |         |        |             |                                     | 52 |
| 53 |         |        |             |                                     | 53 |
| 54 |         |        |             |                                     | 54 |
| 55 |         |        |             |                                     | 55 |
| 56 |         |        |             |                                     | 56 |
| 57 |         |        |             |                                     | 57 |
| 58 |         |        |             |                                     | 58 |
| 59 |         |        |             |                                     | 59 |
| 60 |         |        |             |                                     | 60 |

```
1 # PROGRAM DESCRIPTION: S40.1 DATE: 15 NOV 66
2 # MOD NO2 LOG SECTION P40-P47
3
4 # MOD BY ZELDIN AND ADAPTED BY TALAYCO
5 #
6 # FUNCTIONAL DESCRIPTION
7 # COMPUTE INITIAL THRUST DIRECTION(UT) AND INITIAL VALUE OF VG
8 # VECTOR(VGTIG).
9 #
10 # CALLING SEQUENCE
11 # L CALL
12 # L+1 S40.1
13 #
14 # NORMAL EXIT MODE
15 # AT L+2 OF CALLING SEQUENCE (GOTO L+2) NORMAL RETURN OR
16 # ERROR RETURN IF NOSOFLAG =1
17 #
18 # SUBROUTINES CALLED
19 # LEMPREC
20 # INITVEL
21 # CALCGRAV
22 # MIDGIM
23 #
24 # ALARM OR ABORT EXIT MODES
25 # L+2 OF CALLING SEQUENCE, UNSOLVABLE CONIC IF NOSOFLAG=1
26 #
27 # ERASABLE INITIALIZATION REQUIRED
28 # WEIGHT/G ANTICIPATED VEHICLE MASS DP B16 KGM
29 # XDELVFLG 1=DELTA-V MANEUVER, 0=AIMPT STEER
30 # F THRUST FOR ENGINE USED
31 # IF DELTA-V MANEUVER:
32 # DELVSIN SPECIFIED DELTA-V REQUIRED IN
33 # INERTIAL COORDS. OF ACTIVE VEHICLE
34 # AT TIME OF IGNITION VECTOR B7 M/CS
35 # DELVSAB MAG. OF DELVSIN DP B7 M/CS
36 # RTIG POSITION AT TIME OF IGNITION VECTOR B29 M
37 # VTIG VELOCITY AT TIME OF IGNITION VECTOR B7 M/CS.
38 # IF AIMPT STEER:
39 # TIG TIME OF IGNITION DP B28 CS
40 # RTARG POSITION TARGET TIME VECTOR B29 M
41 # CSTEER C FOR STEER LAW DP B2
42 # DLTARG TARGET TIME-IGNITION TIME DP B28 CS
43 #
44 # OUTPUT
45 # UT DESIRED THRUST DIRECTION VECT. B2 M/(CS.CS)
46 # VGTIG INITIAL VALUE OF VELOCITY
47 # TO BE GAINED (INERT. COORD.) VECTOR B7 M/CS
48 # DELVLVC VGTIG IN LOC. VERT. COORDS. B7 M/CS
49 # BDT V REQUIRED AT TIG -V REQUIRED AT (TIG-2SEC)
50 # -GDT FOR S40.13 VECT B7 M/CS
51 # RTIG CALC IN S40.1B (AIMPT) FOR S40.2,3 VECTOR B27M
52 # POSITION AT TIME OF IGNITION
53 #
54 # DEBRIS QTEMP1
55 # MPAC, QPRET
56 # PUSHLIST
57 #
58 # BANK 14
59 # SETLOC P40S1
60 # BANK
```



|    |           |        |                                           |            |    |
|----|-----------|--------|-------------------------------------------|------------|----|
| 1  | # 10 1 11 |        | PAGE 109                                  |            | 1  |
| 2  | S40.1     | COUNT* | \$\$/S40.1                                |            | 2  |
| 3  |           | STQ    | DLOAD                                     |            | 3  |
| 4  |           |        | QTEMP                                     |            | 4  |
| 5  |           |        | TIG                                       |            | 5  |
| 6  |           | STORE  | TIGSAVE                                   |            | 6  |
| 7  | DELVTEST  | BOFF   |                                           |            | 7  |
| 8  |           |        | XDELVFLG                                  |            | 8  |
| 9  |           |        | S40.1B                                    |            | 9  |
| 10 | CALCTHET  | SETPD  | VLOAD                                     |            | 10 |
| 11 |           |        | 0                                         |            | 11 |
| 12 |           |        | VTIG                                      |            | 12 |
| 13 |           | STORE  | VINIT                                     |            | 13 |
| 14 |           | VXV    | UNIT                                      |            | 14 |
| 15 |           |        | RTIG                                      |            | 15 |
| 16 |           | STOVL  | UT                                        | # UP IN UT | 16 |
| 17 |           |        | RTIG                                      |            | 17 |
| 18 |           | STORE  | RINIT                                     |            | 18 |
| 19 |           | VSQ    | PDDL                                      |            | 19 |
| 20 |           |        | 36D                                       |            | 20 |
| 21 |           | DMP    | DDV                                       |            | 21 |
| 22 |           |        | THETACON                                  |            | 22 |
| 23 |           | DMP    | DMP                                       |            | 23 |
| 24 |           |        | DELVSAB                                   |            | 24 |
| 25 |           |        | WEIGHT/G                                  |            | 25 |
| 26 |           | DDV    |                                           |            | 26 |
| 27 |           |        | F                                         |            | 27 |
| 28 |           | STOVL  | 14D                                       |            | 28 |
| 29 |           |        | DELVSIN                                   |            | 29 |
| 30 |           |        |                                           |            | 30 |
| 31 |           | DOT    | VXSC                                      |            | 31 |
| 32 |           |        | UT                                        |            | 32 |
| 33 |           |        | UT                                        |            | 33 |
| 34 |           | VSL2   | PUSH                                      |            | 34 |
| 35 |           | BVSU   | # (DELTA V.UP)UP SCALED AT 2(+7) P.D.L. 0 |            | 35 |
| 36 |           |        | # DELTA VP SCALED AT 2(+7) P.D.L. 6       |            | 36 |
| 37 |           |        | PDDL                                      |            | 37 |
| 38 |           |        | DELVSIN                                   |            | 38 |
| 39 |           | SIN    | 14D                                       |            | 39 |
| 40 |           |        | PDVL                                      |            | 40 |
| 41 |           |        | 6D                                        |            | 41 |
| 42 |           | VXV    | UNIT                                      |            | 42 |
| 43 |           |        | UT                                        |            | 43 |
| 44 |           | VXSC   | STADR                                     |            | 44 |
| 45 |           | STOVL  | VGTIG                                     |            | 45 |
| 46 |           | UNIT   | # UNIT(VP X UP)SIN(THETAT/2) IN VGTIG.    |            | 46 |
| 47 |           |        | # UNIT(DELTA VP) IN P.D.L. 6              |            | 47 |
| 48 |           |        | PDDL                                      |            | 48 |
| 49 |           |        | 14D                                       |            | 49 |
| 50 |           | COS    | VXSC                                      |            | 50 |
| 51 |           | VAD    | VXSC                                      |            | 51 |
| 52 |           |        | VGTIG                                     |            | 52 |
| 53 |           |        | 36D                                       |            | 53 |
| 54 |           | VSL2   | VAD                                       |            | 54 |
| 55 |           | STADR  |                                           |            | 55 |
| 56 |           |        |                                           |            | 56 |
| 57 |           |        |                                           |            | 57 |
| 58 |           |        |                                           |            | 58 |
| 59 |           |        |                                           |            | 59 |
| 60 |           |        |                                           |            | 60 |

|          |        |                  |                                              |
|----------|--------|------------------|----------------------------------------------|
|          | STORE  | VG TIG           | # VG IGNITION SCALED AT 2(+7) M/CS           |
|          | UNIT   | UT               | # THRUST DIRECTION SCALED AT 2(+1)           |
|          | STOVL  | VG TIG           |                                              |
|          | PUSH   | CALL             |                                              |
|          |        | GET.LVC          | # VG TIG IN LV COOR AT 2(+7) M/CS IN DELVLVC |
|          | GOTO   |                  |                                              |
| S40.1B   | DLOAD  | QTEMP            |                                              |
|          |        | TIG              |                                              |
|          | STORE  | TDEC1            |                                              |
|          | BDSU   |                  |                                              |
|          |        | TPASS4           |                                              |
|          | STCALL | DELLT4           | # INTERCEPT TIME -- TIG.                     |
|          |        | LEMPREC          |                                              |
|          | VLOAD  | SETPD            | # LOAD STATE VECTOR AT TIG FOR INITVEL.      |
|          |        | RATT             |                                              |
|          |        | 0                |                                              |
|          | STORE  | RTIG             |                                              |
|          | STORE  | RINIT            |                                              |
|          | UNIT   |                  |                                              |
|          | STOVL  | UNIT/R/          |                                              |
|          |        | VATT             |                                              |
|          | STORE  | VTIG             |                                              |
|          | STORE  | VINIT            |                                              |
|          | DLOAD  | PDDL             | # NUMIT = 0                                  |
|          |        | ZEROVECS         |                                              |
|          |        | EPS1             |                                              |
|          | BOFF   | DAD              |                                              |
|          |        | NORMSW           |                                              |
|          |        | SMALLEPS         |                                              |
|          |        | EPS2             | # EPSILON4 = 10 DEGREES OR 45 DEGREES.       |
| SMALLEPS | PUSH   | SXA,1            |                                              |
|          |        | RTX1             |                                              |
|          | SXA,2  | CALL             |                                              |
|          |        | RTX2             |                                              |
|          |        | INITVEL          |                                              |
|          | VLOAD  | PUSH             |                                              |
|          |        | DELVEET3         | # VG TIG = VR - VN.                          |
|          | STORE  | VG TIG           |                                              |
|          | UNIT   |                  | # UT = UNIT (VG TIG)                         |
|          | STODL  | UT               |                                              |
|          |        | 36D              |                                              |
|          | STCALL | VGDISP           | # CONVERT VG TIG (IN PUSHLIST) TO LOCAL      |
|          |        | GET.LVC          | # VERTICAL COORDINATES.                      |
|          | GOTO   |                  |                                              |
|          |        | QTEMP            |                                              |
| EPS1     | 2DEC*  | 2.777777778 E-2* | # 10 DEGREES AT 1 REVOLUTION                 |



|    |          |       |                  |                               |    |
|----|----------|-------|------------------|-------------------------------|----|
| 1  |          |       |                  |                               | 1  |
| 2  | EPS2     | 2DEC* | 9.722222222 E-2* | # 35 DEGREES AT 1 REVOLUTION. | 2  |
| 3  |          |       |                  |                               | 3  |
| 4  | THETACON | 2DEC  | .31830989 B-8    |                               | 4  |
| 5  |          |       |                  |                               | 5  |
| 6  |          |       |                  |                               | 6  |
| 7  |          |       |                  |                               | 7  |
| 8  |          |       |                  |                               | 8  |
| 9  |          |       |                  |                               | 9  |
| 10 |          |       |                  |                               | 10 |
| 11 |          |       |                  |                               | 11 |
| 12 |          |       |                  |                               | 12 |
| 13 |          |       |                  |                               | 13 |
| 14 |          |       |                  |                               | 14 |
| 15 |          |       |                  |                               | 15 |
| 16 |          |       |                  |                               | 16 |
| 17 |          |       |                  |                               | 17 |
| 18 |          |       |                  |                               | 18 |
| 19 |          |       |                  |                               | 19 |
| 20 |          |       |                  |                               | 20 |
| 21 |          |       |                  |                               | 21 |
| 22 |          |       |                  |                               | 22 |
| 23 |          |       |                  |                               | 23 |
| 24 |          |       |                  |                               | 24 |
| 25 |          |       |                  |                               | 25 |
| 26 |          |       |                  |                               | 26 |
| 27 |          |       |                  |                               | 27 |
| 28 |          |       |                  |                               | 28 |
| 29 |          |       |                  |                               | 29 |
| 30 |          |       |                  |                               | 30 |
| 31 |          |       |                  |                               | 31 |
| 32 |          |       |                  |                               | 32 |
| 33 |          |       |                  |                               | 33 |
| 34 |          |       |                  |                               | 34 |
| 35 |          |       |                  |                               | 35 |
| 36 |          |       |                  |                               | 36 |
| 37 |          |       |                  |                               | 37 |
| 38 |          |       |                  |                               | 38 |
| 39 |          |       |                  |                               | 39 |
| 40 |          |       |                  |                               | 40 |
| 41 |          |       |                  |                               | 41 |
| 42 |          |       |                  |                               | 42 |
| 43 |          |       |                  |                               | 43 |
| 44 |          |       |                  |                               | 44 |
| 45 |          |       |                  |                               | 45 |
| 46 |          |       |                  |                               | 46 |
| 47 |          |       |                  |                               | 47 |
| 48 |          |       |                  |                               | 48 |
| 49 |          |       |                  |                               | 49 |
| 50 |          |       |                  |                               | 50 |
| 51 |          |       |                  |                               | 51 |
| 52 |          |       |                  |                               | 52 |
| 53 |          |       |                  |                               | 53 |
| 54 |          |       |                  |                               | 54 |
| 55 |          |       |                  |                               | 55 |
| 56 |          |       |                  |                               | 56 |
| 57 |          |       |                  |                               | 57 |
| 58 |          |       |                  |                               | 58 |
| 59 |          |       |                  |                               | 59 |
| 60 |          |       |                  |                               | 60 |

```

1 # SUBROUTINE NAME: S40.2,3 MOD. NO. 3, DATE APRIL 4, 1967
2 # MODIFICATION BY: JONATHON D. ADDELSTON (ADAMS ASSOCIATES)
3
4 # MOD. NO. 4: JULY 18, 1967: PETER ADLER (MIT/IL)
5 # MOD. NO. 5: OCTOBER 18, 1967: PETER ADLER (MIT/IL)
6 # ORIGINALLY BY: SAYDEAN ZELDIN (MIT INSTRUMENTATION LAB) AND RICHARD TALAYCO (SYSTEM DELVELOPMENT CORP)
7
8 # S40.2,3 COMPUTES "POINTVSM" WHICH IS THE HALF-UNIT DESIRED THRUST VECTOR IN STABLE-MEMBER COORDINATES FROM "UT"
9 # WHICH IS THE SAME VECTOR IN REFERENCE COORDINATES. IT DETERMINES THE CORRECT VALUES FOR "SCAXIS" USING THE +X
10 # AXIS FOR DPS, APS, AND RCS BURNS. THE "WINGS-LEVEL HEADS-UP" LM ORIENTATION IS THEN COMPUTED IN REFERENCE
11 # COORDINATES. THESE VECTORS ALSO DEFINE THE "PREFERRED IMU ORIENTATION". UPON COMPLETION OF THIS CALCULATION,
12 # THE "PREFERRED ATTITUDE COMPUTED" FLAG IS SET (PFRATFLG).
13
14 # CALLING SEQUENCE:
15 # L CALL # INTERPRETIVE CALL.
16 # L +1 S40.2,3
17 # L +2 (RETURN) # GIMBAL ANGLE VECTOR IN MPAC.
18
19 # SUBROUTINES CALLED: NONE.
20
21 # NORMAL RETURN: L +2 (SEE CALLING SEQUENCE ABOVE).
22
23 # ALARM/ABORT MODES: NONE.
24
25 # INPUT:
26 # 1. REFSMMAT MATRIX FROM REFERENCE TO STABLE-MEMBER COORDINATES SCALED AT 2.
27 # 2. UT HALF-UNIT DESIRED THRUST DIRECTION.
28 # 3. RTIG POSITION AT TIG IN REFERENCE COORDINATES.
29
30 # OUTPUT:
31 # 1. `XSCREF' WINGS-LEVEL HEADS-UP LM ORIENTATION
32 # `YSCREF' IN REFERENCE COORDINATES
33 # `ZSCREF' (PREFERRED IMU ORIENTATION).
34 # 2. POINTVSM DESIRED THRUST DIRECTION IN STABLE-MEMBER COORDINATES.
35 # 3. SCAXIS HALF-UNIT OF AXIS TO ALIGN IN STABLE-MEMBER COORDINATES.
36 # 4. PFRATFLG INTERPRETIVE FLAG. ON: PREFERRED ORIENTATION COMPUTED; OFF: NOT COMPUTED.
37
38 # DEBRIS: NONE

```

|    |           |        |            |                                            |
|----|-----------|--------|------------|--------------------------------------------|
| 1  | # P40 P41 |        |            |                                            |
| 2  |           |        |            |                                            |
| 3  |           | COUNT* | \$\$/S40.2 |                                            |
| 4  | S40.2,3   | VLOAD  | UT         | # UT: DESIRED THRUST DIRECTION (HALF-UNIT) |
| 5  |           |        |            | # (PUT INTO TOP OF PUSH-DOWN-LIST.)        |
| 6  |           | MXV    | VSL1       | # TRANSFORM THRUST DIRECTION TO STABLE-    |
| 7  |           |        | REFSMMAT   | # MEMBER FROM REFERENCE COORDS (RESCALE).  |
| 8  |           | STOVL  | POINTVSM   | # SAVE FOR "VECPPOINT" ROUTINE (LEMMANU).  |
| 9  |           |        | UNITX      | # SCAXIS SET TO +X, FOR P40 AND P42 AND    |
| 10 |           | STOVL  | SCAXIS     | # FOR P41 IF RCS NOT -X,+Y,-Y,+Z,-Z.       |
| 11 |           |        | UT         | # ASSUME +X BURN ALWAYS, EVEN FOR RCS.     |
| 12 | PLUSX     | STORE  | XSCREF     | # XSCREF = UT (DESIRED THRUST DIRECTION)   |
| 13 |           | VXV    | UNIT       | # RTIG = POSITION AT TIME-OF-IGNITION.     |
| 14 |           |        | RTIG       | # YSCREF = UNIT(UT X RTIG)                 |
| 15 |           | PDDL   | BHIZ       |                                            |
| 16 |           |        | 36D        | # TEST MAGNITUDE OF UT X RTIG              |
| 17 |           |        | FIXY       | # IF SMALL, USE UT X VTIG AS YSC           |
| 18 | STORY     | VLOAD  | STADR      |                                            |
| 19 |           | STORE  | YSCREF     |                                            |
| 20 |           | VXV    | VSL1       | # COMPUTE (YSCREF X XSCREF), BUT FOR A     |
| 21 |           |        | XSCREF     | # RIGHT HANDED SYSTEM, NEED (X CROSS Y).   |
| 22 |           | VCOMP  |            | # ZSCREF = - (YSCREF X XSCREF)             |
| 23 |           | STORE  | ZSCREF     | # = + (XSCREF X YSCREF)                    |
| 24 |           |        |            |                                            |
| 25 |           | SET    | RVQ        |                                            |
| 26 |           |        | PFRATFLG   |                                            |
| 27 | FIXY      | VLOAD  | VXV        | # IN THIS CASE,                            |
| 28 |           |        | XSCREF     | # YSCREF = UNIT(XSCREF X VTIG)             |
| 29 |           |        | VTIG       |                                            |
| 30 |           | UNIT   | PUSH       |                                            |
| 31 |           | GOTO   |            |                                            |
| 32 |           |        | STORY      |                                            |
| 33 |           |        |            |                                            |



```

1 # SUBROUTINE S40.8
2 # MODIFIED APRIL 3, 1968 BY PETER ADLER, MIT/IL
3
4 #
5 # DESCRIPTION
6 # S40.8 UPDATES THE VELOCITY-TO-BE-GAINED VECTOR, VG, (AND FOR LAMBERT TARGETTED BURNS ALSO EXTRAPOLATES VG
7 # USING THE BDT VECTOR) COMPUTES THE TIME FOR ISSUING THE ENGINE OFF COMMAND, TGO, AND CALLS THE ROUTINE
8 # "FINDCDUW", WHICH GENERATES STEERING COMMANDS FOR THE DAP.
9 #
10 # CALLING SEQUENCE
11 # L-1 CALL
12 # L S40.8
13 # L+1 INTERPRETIVE RETURN
14 #
15 # ALARM
16 # IF VG . DELVREF IS NEGATIVE (VG AND DELVREF OVER 90 DEGREES APART), BYPASS TGO AND STEERING COMPUTATIONS
17 # AND SET ALARM 1407. RETURN TO CALLER NORMALLY.
18 #
19 # INPUT AND INITIALIZATION
20 # VGPREV REFERENCE 2(7) M/CS
21 # DELVREF REFERENCE 2(7) M/CS
22 # BDT REFERENCE 2(7) M/CS
23 # TDECAY TAIL-OFF TIME 2(28) CS
24 # XDELVFLG 1 = EXTERNAL DELTA-V; 0 = LAMBERT (AIMPOINT)
25 # STEERSW 1 = DO STEERING AND TGO COMPUTATIONS; 0 = VG UPDATE ONLY
26 # FIRSTFLG 1 = GONE TO LAMBERT AT LEAST ONCE; 0 = HAVEN'T GONE TO LAMBERT YET.
27 #
28 # NOTE: VGTIG EQUALS VGPREV
29 #
30 # OUTPUT
31 # STEERSW SEE INPUT
32 # INPULSW 1 = ENGINE OFF IN TGO CENTISECONDS; 0 = CONTINUE BURN
33 # TGO TIME TO CUT-OFF 2(28) CS
34 # SEE FINDCDUW FOR STEERING OUTPUTS.
35 #
36 # SUBROUTINE CALLED
37 # FINDCDUW
38 #
39 # DEBRIS
40 # MPACS, PUSHLIST
41
42 COUNT* $$/S40.8
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

```

|    |           |       |                                            |    |
|----|-----------|-------|--------------------------------------------|----|
| 1  | # 1000000 |       |                                            | 1  |
| 2  | S40.8     | BOF   | # GENERATE VR IF NOT EXTERNAL DELTA-V BURN | 2  |
| 3  |           |       | XDELVFLG                                   | 3  |
| 4  |           | VLOAD | RASTEER1                                   | 4  |
| 5  |           |       | VSU                                        | 5  |
| 6  |           |       | VGPREV                                     | 6  |
| 7  | VGAIN*    | STORE | DELVREF                                    | 7  |
| 8  |           | MXV   | VG                                         | 8  |
| 9  |           |       | VSL1                                       | 9  |
| 10 |           |       | # VELOCITY TO BE GAINED SCALED AT (7) M/CS | 10 |
| 11 | BDTOK     | STORE | REFSMMAT                                   | 11 |
| 12 |           | VLOAD | UNFC/2                                     | 12 |
| 13 |           |       | ABVAL                                      | 13 |
| 14 | TGDCALC   | STORE | VG                                         | 14 |
| 15 |           | SETPD | VGDISP                                     | 15 |
| 16 |           |       | VLOAD                                      | 16 |
| 17 |           |       | 0                                          | 17 |
| 18 |           | STOVL | VG                                         | 18 |
| 19 |           |       | VGPREV                                     | 19 |
| 20 |           | BOFF  | DELVREF                                    | 20 |
| 21 |           |       | VCOMP                                      | 21 |
| 22 |           |       | STEERSW                                    | 22 |
| 23 |           | UNIT  | QPRET                                      | 23 |
| 24 |           | DOT   | PUSH                                       | 24 |
| 25 |           |       | VG                                         | 25 |
| 26 |           | BPL   | DDV                                        | 26 |
| 27 |           |       | ALARMIT                                    | 27 |
| 28 |           |       | # DELV IS MORE THAN 90 DEGREES FROM VG.    | 28 |
| 29 |           | DAD   | VEX                                        | 29 |
| 30 |           |       | DMP                                        | 30 |
| 31 |           | SR    | DPHALF                                     | 31 |
| 32 |           |       | DDV                                        | 32 |
| 33 |           |       | 10D                                        | 33 |
| 34 |           |       | 36D                                        | 34 |
| 35 |           | DMP   | DAD                                        | 35 |
| 36 |           |       | -FOURDT                                    | 36 |
| 37 |           | STORE | TDECAY                                     | 37 |
| 38 |           | DAD   | TGO                                        | 38 |
| 39 |           |       | PIPTIME                                    | 39 |
| 40 |           | STODL | TIG                                        | 40 |
| 41 |           |       | TGO                                        | 41 |
| 42 |           | DSU   | BPL                                        | 42 |
| 43 |           |       | FOURSECS                                   | 43 |
| 44 |           |       | # 400 CS                                   | 44 |
| 45 |           | SET   | FINDCDUW -2                                | 45 |
| 46 |           |       | CLRGO                                      | 46 |
| 47 |           |       | IMPULSW                                    | 47 |
| 48 |           |       | STEERSW                                    | 48 |
| 49 |           |       | QPRET                                      | 49 |
| 50 | ALARMIT   | EXIT  |                                            | 50 |
| 51 |           |       |                                            | 51 |
| 52 |           |       |                                            | 52 |
| 53 |           |       |                                            | 53 |
| 54 |           |       |                                            | 54 |
| 55 |           |       |                                            | 55 |
| 56 |           |       |                                            | 56 |
| 57 |           |       |                                            | 57 |
| 58 |           |       |                                            | 58 |
| 59 |           |       |                                            | 59 |
| 60 |           |       |                                            | 60 |

[illegible]

```

NAME: S40.13 -- TIMEBURN
#
FUNCTION (1) DETERMINE WHETHER A GIVEN COMBINATION OF VELOCITY TO
BE GAINED AND ENGINE CHOICE RESULT IN A BURN TIME
SUFFICIENT TO ALLOW STEERING AT THE VEHICLE DURING THE
BURN
(2) THE MAGNITUDE OF THE RESULTING BURN TIME -- IF IT
IS SHORT -- AND THE ASSOCIATED TIME OF THE ENGINE OFF
SIGNAL
CALLING SEQUENCE VIA FINDVAC AS A NEW JOB
#
INPUT VGTIG -- VELOCITY TO BE GAINED VECTOR (METERS/CS) AT +7
WEIGHT/G -- MASS OF VEHICLE IN KGM AT +16
F -- APS ENGINE THRUST IN M.NEWTONS AT +7
AND ALSO FOR RCS ENGINE
MDOT -- RATE OF DECREASE OF VEHICLE MASS DURING ENGINE
BURN IN KILOGRAMS/CS AT +3. THIS SCALING MAY
REQUIRE MODIFICATION FOR SATURN BURNS.
ENGLFLAG -- SWITCH TO DECIDE WHETHER APS OR DPS ENGINE IS USED
=0 DPS
=1 APS
#
OUTPUT IMPULSW ZERO FOR STEERING
ONE FOR ATTITUDE HOLD
NOTHROTL ZERO FOR THROTTLING
ONE TO INHIBIT THROTTLING
TGO TIME TO BURN IN CS
#
THE QUANTITY M.NEWTON = 10000 NEWTONS WILL BE USED TO EXPRESS
FORCE.
#
EBANK= TGO
COUNT* $$/40.13
TC INTPRET
SETPD CLEAR
OOD
IMPULSW # ASSUME NO STEERING UNTIL FOUND OTHERWISE
#
VLOAD ABVAL
VGTIG # VELOCITY TO BE GAINED AT +7
PDDL DMP # OOD = MAG OF VGTIG AT +7
4SEC(17) # CORRECT VG FOR 4 SECS OF 2 JET ULLAGE
FRCS2
DDV SL1 # SCALE
#
BDSU WEIGHT/G
BOFF PUSH
SET
APSFLAG
S40.13D # FOR DPS ENGINE
NOTHROTL
#
DLOAD DDV # OOD = MAG OF VGTIG CORRECTED
K1VAL # M.NEWTONS-CS AT +24
WEIGHT/G
BDSU BMN

```

|  |     |          |
|--|-----|----------|
|  |     | S40.136  |
|  |     | NOTHROTL |
|  | BOV | PUSH     |

|    |  |
|----|--|
| 1  |  |
| 2  |  |
| 3  |  |
| 4  |  |
| 5  |  |
| 6  |  |
| 7  |  |
| 8  |  |
| 9  |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |
| 21 |  |
| 22 |  |
| 23 |  |
| 24 |  |
| 25 |  |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 |  |
| 31 |  |
| 32 |  |
| 33 |  |
| 34 |  |
| 35 |  |
| 36 |  |
| 37 |  |
| 38 |  |
| 39 |  |
| 40 |  |
| 41 |  |
| 42 |  |
| 43 |  |
| 44 |  |
| 45 |  |
| 46 |  |
| 47 |  |
| 48 |  |
| 49 |  |
| 50 |  |
| 51 |  |
| 52 |  |
| 53 |  |
| 54 |  |
| 55 |  |
| 56 |  |
| 57 |  |
| 58 |  |
| 59 |  |
| 60 |  |
| 61 |  |
| 62 |  |
| 63 |  |
| 64 |  |
| 65 |  |
| 66 |  |
| 67 |  |
| 68 |  |
| 69 |  |
| 70 |  |
| 71 |  |
| 72 |  |
| 73 |  |
| 74 |  |
| 75 |  |
| 76 |  |
| 77 |  |
| 78 |  |
| 79 |  |
| 80 |  |

|    |   |
|----|---|
| 76 | 1 |
| 77 |   |







|    |             |       |                                             |    |
|----|-------------|-------|---------------------------------------------|----|
| 1  | # 140 1 1 1 |       |                                             | 1  |
| 2  | GETVRVG1    | VXSC  | VAD                                         | 2  |
| 3  |             |       | 32D                                         | 3  |
| 4  | GETVRVG2    | LXC,2 | VSR*                                        | 4  |
| 5  |             |       | RTX2                                        | 5  |
| 6  |             |       | 0 -1,2                                      | 6  |
| 7  |             | STORE | VIPRIME                                     | 7  |
| 8  |             | GOTO  |                                             | 8  |
| 9  |             |       | ASTREND -2                                  | 9  |
| 10 | 180MESS     | VLOAD | DOT                                         | 10 |
| 11 |             |       | IC                                          | 11 |
| 12 |             |       | UNIT/R/                                     | 12 |
| 13 |             | BMN   | VLOAD                                       | 13 |
| 14 |             |       | NEGPROD                                     | 14 |
| 15 |             |       | IC                                          | 15 |
| 16 |             | VSR1  | PDVL                                        | 16 |
| 17 |             |       | UNIT/R/                                     | 17 |
| 18 |             | VSR1  | VAD                                         | 18 |
| 19 |             | UNIT  |                                             | 19 |
| 20 |             | PUSH  | VCOMP # FOR A                               | 20 |
| 21 |             | VXV   | SIGN                                        | 21 |
| 22 |             |       | UN                                          | 22 |
| 23 |             |       | GEOMSGN                                     | 23 |
| 24 |             | UNIT  | VXSC                                        | 24 |
| 25 |             |       | 30D                                         | 25 |
| 26 |             | PDVL  | # UNIT(IC-IR) +-B                           | 26 |
| 27 |             | GOTO  |                                             | 27 |
| 28 | NEGPROD     | VLOAD | GETVRVG1                                    | 28 |
| 29 |             |       | VSR1                                        | 29 |
| 30 |             |       | UNIT/R/                                     | 30 |
| 31 |             | PDVL  | VSR1                                        | 31 |
| 32 |             |       | IC                                          | 32 |
| 33 |             | VSU   | UNIT                                        | 33 |
| 34 |             | PUSH  |                                             | 34 |
| 35 |             | VXV   | SIGN                                        | 35 |
| 36 |             |       | UN # FOR B                                  | 36 |
| 37 |             |       | GEOMSGN                                     | 37 |
| 38 |             | UNIT  | VXSC                                        | 38 |
| 39 |             |       | 32D                                         | 39 |
| 40 |             | PDVL  |                                             | 40 |
| 41 |             | VXSC  | VAD                                         | 41 |
| 42 |             |       | 30D                                         | 42 |
| 43 |             | GOTO  |                                             | 43 |
| 44 |             |       | GETVRVG2                                    | 44 |
| 45 |             | VSU   |                                             | 45 |
| 46 | ASTREND     | STORE | VN1                                         | 46 |
| 47 | FIRSTTME    | SLOAD | DELVEET3                                    | 47 |
| 48 |             |       | BZE                                         | 48 |
| 49 |             |       | RTX2                                        | 49 |
| 50 |             |       | GETGOBL                                     | 50 |
| 51 |             | VLOAD | GOTO # NO OBLATENESS COMP IF IN MOON SPHERE | 51 |
| 52 |             |       |                                             | 52 |
| 53 |             |       |                                             | 53 |
| 54 |             |       |                                             | 54 |
| 55 |             |       |                                             | 55 |
| 56 |             |       |                                             | 56 |
| 57 |             |       |                                             | 57 |
| 58 |             |       |                                             | 58 |
| 59 |             |       |                                             | 59 |
| 60 |             |       |                                             | 60 |



|    |         |       |      |    |
|----|---------|-------|------|----|
| 1  |         |       |      | 1  |
| 2  |         |       |      | 2  |
| 3  |         |       |      | 3  |
| 4  | GETGOBL | VLOAD | UNIT | 4  |
| 5  |         |       |      | 5  |
| 6  |         |       |      | 6  |
| 7  |         |       |      | 7  |
| 8  |         |       |      | 8  |
| 9  |         |       |      | 9  |
| 10 |         |       |      | 10 |
| 11 |         |       |      | 11 |
| 12 |         |       |      | 12 |
| 13 |         |       |      | 13 |
| 14 |         |       |      | 14 |
| 15 |         |       |      | 15 |
| 16 |         |       |      | 16 |
| 17 |         |       |      | 17 |
| 18 |         |       |      | 18 |
| 19 |         |       |      | 19 |
| 20 |         |       |      | 20 |
| 21 |         |       |      | 21 |
| 22 |         |       |      | 22 |
| 23 |         |       |      | 23 |
| 24 |         |       |      | 24 |
| 25 |         |       |      | 25 |
| 26 |         |       |      | 26 |
| 27 |         |       |      | 27 |
| 28 |         |       |      | 28 |
| 29 |         |       |      | 29 |
| 30 |         |       |      | 30 |
| 31 |         |       |      | 31 |
| 32 |         |       |      | 32 |
| 33 |         |       |      | 33 |
| 34 |         |       |      | 34 |
| 35 |         |       |      | 35 |
| 36 |         |       |      | 36 |
| 37 |         |       |      | 37 |
| 38 |         |       |      | 38 |
| 39 |         |       |      | 39 |
| 40 |         |       |      | 40 |
| 41 |         |       |      | 41 |
| 42 |         |       |      | 42 |
| 43 |         |       |      | 43 |
| 44 |         |       |      | 44 |
| 45 |         |       |      | 45 |
| 46 |         |       |      | 46 |
| 47 |         |       |      | 47 |
| 48 |         |       |      | 48 |
| 49 |         |       |      | 49 |
| 50 |         |       |      | 50 |
| 51 |         |       |      | 51 |
| 52 |         |       |      | 52 |
| 53 |         |       |      | 53 |
| 54 |         |       |      | 54 |
| 55 |         |       |      | 55 |
| 56 |         |       |      | 56 |
| 57 |         |       |      | 57 |
| 58 |         |       |      | 58 |
| 59 |         |       |      | 59 |
| 60 |         |       |      | 60 |

```
1 # TRIMGIMB (FORMERLY S40.6)
2 # MOD 0 24 FEB 67 PETER ADLER
3
4 #
5 # FUNCTION:
6 # TRIMS DPS ENGINE TO MINIMIZE THRUST/CG OFFSET. ENGINE IS GIMBALLED TO FULL + PITCH AND + ROLL (TO LOCK)
7 # FOR REFERENCE AND IS THEN BROUGHT BACK TO TRIM POSITION BY RUNNING FOR THE PROPER TIMES (TO BE
8 # SPECIFIED BY GAEC) IN - PITCH AND - ROLL.
9 #
10 # CALLING SEQUENCE:
11 # VIA WAITLIST FROM R03
12 #
13 # INPUT:
14 # PITTIME TIME TO RUN FROM FULL + PITCH TO TRIM (CS)
15 # ROLLTIME TIME TO RUN FROM FULL + ROLL TO TRIM (CS)
16 #
17 # SUBROUTINES USED:
18 # WAITLIST, FIXDELAY, VARDELAY, FLAGUP, FLAGDOWN, NOVAC
19
20 COUNT* $$/S40.6
21 EBANK= ROLLTIME # OCTAL MASKS: PRI05=05000 EBANK5=02400
22
23 TRIMGIMB TC DOWNFLAG # GMBDRVSW FLAG IS SET WHEN EITHER ROLL OR
24 ADRES GMBDRVSW # PITCH IS COMPLETED, WHICHEVER IS FIRST.
25
26 CS PRI05 # TURN OFF - PITCH, - ROLL, IF ON.
27 EXTEND
28 WAND CHAN12
29 CAF EBANK5 # TURN ON + PITCH, + ROLL.
30 EXTEND
31 WOR CHAN12
32 TC FIXDELAY # WAIT ONE MINUTE TO MAKE SURE ENGINE IS
33 DEC 6000 # AT FULL + PITCH AND FULL + ROLL
34 CS EBANK5 # TURN OFF + PITCH, + ROLL.
35 EXTEND
36 WAND CHAN12
37 CAF PRI05 # TURN ON - PITCH, - ROLL.
38 EXTEND
39 WOR CHAN12
40 CAE PITTIME # GET TIME TO SHUT OFF - PITCH AND SET UP
41 TC TWIDDLE # TWIDDLE-TASK TO TURN IT OFF THEN
42 ADRES PITCHOFF
43
44 CAE ROLLTIME # GET TIME TO SHUT OFF - ROLL AND GO AWAY
45 TC VARDELAY # UNTIL THEN
46 CS BIT12
47 EXTEND
48 WAND CHAN12 # SHUT OFF ROLL
49 ROLLOVER CA FLAGWRD6 # IF HERE INLINE (ROLL DONE) IS PITCH DONE
50 MASK GMBDRBIT # IF HERE FROM PITCHOFF, IS ROLL DONE?
51 EXTEND
52 BZF PITCHOFF +4 # NO. SET FLAG, ROLL OR PITCH DONE.
53 CAF PRI010 # RETURN TO R03.
54 TC NOVAC
55 EBANK= WHOCARES
56
57
58
59
60
```

|    |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |
| 8  |
| 9  |
| 10 |
| 11 |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 |
| 17 |
| 18 |
| 19 |
| 20 |
| 21 |
| 22 |
| 23 |
| 24 |
| 25 |
| 26 |
| 27 |
| 28 |
| 29 |
| 30 |
| 31 |
| 32 |
| 33 |
| 34 |
| 35 |
| 36 |
| 37 |
| 38 |
| 39 |
| 40 |
| 41 |
| 42 |
| 43 |
| 44 |
| 45 |
| 46 |
| 47 |
| 48 |
| 49 |
| 50 |
| 51 |
| 52 |
| 53 |
| 54 |
| 55 |
| 56 |
| 57 |
| 58 |
| 59 |
| 60 |

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65
- 66
- 67
- 68
- 69
- 70
- 71
- 72
- 73
- 74
- 75
- 76
- 77
- 78
- 79
- 80

1412THE

1

```
1 # SUBROUTINE NAME: S41.1 MOD. NO. 0 DATE: FEBRUARY 28, 1967
2 # MOD. NO. 1 DATE: JANUARY 23, 1968, BY PETER ADLER (MIT/IL)
3 #
4 #
5 # AUTHOR: JONATHON D. ADDLESTON (ADAMS ASSOCIATES)
6 #
7 # S41.1 PERFORMS THE COORDINATE SYSTEM TRANSFORMATION FROM THE REFERENCE FRAME TO THE BODY OF THE LM.
8 # SPECIFICALLY, IT IS USED TO TRANSFORM A VELOCITY (SCALED AT 2(+7) METERS/CENTISECOND) FROM REFERENCE TO LM AXIS
9 # COORDINATES. FIRST THE VECTOR IS TRANSFORMED TO THE STABLE MEMBER COORDINATES BY THE MATRIX REFSMMAT. THIS
10 # LEAVES THE VECTOR IN MPAC, SCALED AT 2(+8) METERS/CENTISECOND. THEN
11 # THE SUBROUTINE CDUTRIG IS CALLED TO SET UP THE DOUBLE-PRECISION CDU VECTOR ALONG WITH ITS SINES AND COSINES.
12 # THE VECTOR IS THEN TRANSFORMED FROM STABLE MEMBER COORDINATES TO SPACECRAFT (OR LM) COORDINATES BY THE
13 # SUBROUTINE *SMNB*. FINALLY, THE VECTOR IS RESCALED TO 2(+7) METERS/CENTISECOND, AND CONTROL IS RETURNED BO THE
14 # CALLER WITH C(MPAC) = VELOCITY(LM).
15 #
16 # CALLING SEQUENCE:
17 # L VLOAD CALL
18 # L +1 VELOCITY(REF) # SCALED AT 2(+7) M/CS IN REFERENCE COORDS.
19 # L +2 S41.1
20 # L +3 STORE VELOCITY(LM) # SCALED AT 2(+7) M/CS IN LM BODY AXIS SYS.
21 #
22 # SUBROUTINES CALLED:
23 # 1. CDUTRIG,
24 # WHICH CALLS CDULOGIC.
25 # 2. *SMNB*
26 #
27 # NORMAL RETURN: L +3 (SEE CALLING SEQUENCE, ABOVE.)
28 #
29 # ALARM/ABORT MODES: NONE.
30 #
31 # RESTART PROTECTION: NONE.
32 #
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

# P40-P47

|    |                          |        |                         |                                                   |    |
|----|--------------------------|--------|-------------------------|---------------------------------------------------|----|
| 1  |                          |        |                         |                                                   | 1  |
| 2  | # INPUT:                 |        |                         |                                                   | 2  |
| 3  | #                        | 1.     | REFSMMAT.               |                                                   | 3  |
| 4  | #                        | 2.     | CDUX, CDUY, CDUZ.       |                                                   | 4  |
| 5  | #                        | 3.     | VELOCITY (REF) IN MPAC. |                                                   | 5  |
| 6  | #                        |        |                         |                                                   | 6  |
| 7  | # OUTPUT:                |        |                         |                                                   | 7  |
| 8  | #                        | 1.     | CSUSPOT:                | DOUBLE PRECISION CDU VECTOR, ORDERED Y,Z,X.       | 8  |
| 9  | #                        | 2.     | SINCDU:                 | HALF SINES OF CDUSPOT COMPONENTS                  | 9  |
| 10 | #                        | 3.     | COSCDU:                 | HALF COSINES OF CDUSPOT COMPONENTS.               | 10 |
| 11 | #                        | 4.     | MPAC:                   | VELOCITY(LM) (SCALED AT 2(+7) METERS/CENTISECOND) | 11 |
| 12 | #                        |        |                         |                                                   | 12 |
| 13 | # DEBRIS: NONE.          |        |                         |                                                   | 13 |
| 14 | #                        |        |                         |                                                   | 14 |
| 15 | # CHECKOUT STATUS: CODED |        |                         |                                                   | 15 |
| 16 |                          |        |                         |                                                   | 16 |
| 17 |                          | COUNT* | \$\$/S41.1              |                                                   | 17 |
| 18 | S41.1                    | MXV    | VSL1                    | # CONVERT VECTOR IN MPAC FROM REF AT 2(+7)        | 18 |
| 19 |                          |        | REFSMMAT                | # TO SM AND RESCALE DUE TO HALF-UNIT MATRIX       | 19 |
| 20 |                          | GOTO   |                         | # CONVERT TO BODY AT 2(+7) USING PRESENT          | 20 |
| 21 |                          |        | CDU*SMNB                | # CDU ANGLES. CDU*SMNB WILL RETURN                | 21 |
| 22 |                          |        |                         | # VIA RVQ TO THE CALLER OF S41.1.                 | 22 |
| 23 |                          |        |                         |                                                   | 23 |
| 24 |                          |        |                         |                                                   | 24 |
| 25 |                          |        |                         |                                                   | 25 |
| 26 |                          |        |                         |                                                   | 26 |
| 27 |                          |        |                         |                                                   | 27 |
| 28 |                          |        |                         |                                                   | 28 |
| 29 |                          |        |                         |                                                   | 29 |
| 30 |                          |        |                         |                                                   | 30 |
| 31 |                          |        |                         |                                                   | 31 |
| 32 |                          |        |                         |                                                   | 32 |
| 33 |                          |        |                         |                                                   | 33 |
| 34 |                          |        |                         |                                                   | 34 |
| 35 |                          |        |                         |                                                   | 35 |
| 36 |                          |        |                         |                                                   | 36 |
| 37 |                          |        |                         |                                                   | 37 |
| 38 |                          |        |                         |                                                   | 38 |
| 39 |                          |        |                         |                                                   | 39 |
| 40 |                          |        |                         |                                                   | 40 |
| 41 |                          |        |                         |                                                   | 41 |
| 42 |                          |        |                         |                                                   | 42 |
| 43 |                          |        |                         |                                                   | 43 |
| 44 |                          |        |                         |                                                   | 44 |
| 45 |                          |        |                         |                                                   | 45 |
| 46 |                          |        |                         |                                                   | 46 |
| 47 |                          |        |                         |                                                   | 47 |
| 48 |                          |        |                         |                                                   | 48 |
| 49 |                          |        |                         |                                                   | 49 |
| 50 |                          |        |                         |                                                   | 50 |
| 51 |                          |        |                         |                                                   | 51 |
| 52 |                          |        |                         |                                                   | 52 |
| 53 |                          |        |                         |                                                   | 53 |
| 54 |                          |        |                         |                                                   | 54 |
| 55 |                          |        |                         |                                                   | 55 |
| 56 |                          |        |                         |                                                   | 56 |
| 57 |                          |        |                         |                                                   | 57 |
| 58 |                          |        |                         |                                                   | 58 |
| 59 |                          |        |                         |                                                   | 59 |
| 60 |                          |        |                         |                                                   | 60 |

```
1
2 BANK 32
3 SETLOC F2DPS*32
4 BANK
5
6 EBANK= E2DPS
7
8 # *****
9 # P63: THE LUNAR LANDING, BRAKING PHASE
10 # *****
11
12 COUNT* $$/P63
13
14 P63LM TC PHASCHNG
15 OCT 04024
16
17 TC BANKCALL # DO IMU STATUS CHECK ROUTINE R02
18 CADR R02BOTH
19
20 CAF P63ADRES # INITIALIZE WHICH FOR BURNBABY
21 TS WHICH
22
23 CAF DPSTHRSH # INITIALIZE DVMON
24 TS DVTHRUSH
25 CAF FOUR
26 TS DVCNTR
27
28 CS ONE # INITIALIZE WCHPHASE AND FLPASS0
29 TS WCHPHASE
30
31 CA ZERO
32 TS FLPASS0
33
34 CS BIT14
35 EXTEND
36 WAND CHAN12 # REMOVE TRACK-ENABLE DISCRETE.
37
38 FLAGORGY TC INTERPRET # DIONYSIAN FLAG WAVING
39 CLEAR
40 NOTHROTL
41 REDFLAG
42 CLEAR SET
43 LRBYPASS
44 MUNFLAG
45 CLEAR
46 P25FLAG # TERMINATE P25 IF IT IS RUNNING.
47 RNDVZFLG # TERMINATE P20 IF IT IS RUNNING
48
49 # *****
50
51 IGNALG SETPD VLOAD # FIRST SET UP INPUTS FOR RP-TO-R:-
```



```
1
2 0 # AT 0D LANDING SITE IN MOON FIXED FRAME
3 RLS # AT 6D ESTIMATED TIME OF LANDING
4 PDDL PUSH # MPAC NON-ZERO TO INDICATE LUNAR CASE
5 TLAND
6 STCALL TPIP # ALSO SET TPIP FOR FIRST GUIDANCE PASS
7
8 VSL4 RP-TO-R
9 MXV
10 REFSMMAT
11
12 STCALL LAND
13 GUIDINIT # GUIDINIT INITIALIZES WM AND /LAND/
14 DLOAD DSU
15
16 TLAND
17 GUIDDURN
18 TDEC1 # INTEGRATE STATE FORWARD TO THAT TIME
19
20 SSP LEMPREC
21 VLOAD
22 NIGNLOOP
23
24 40D
25 UNITX
26 CG
27 STOVL
28
29 UNITY
30 CG +6
31 UNITZ
32
33 STODL CG +14
34 99999CON
35 DELTAH # INITIALIZE DELTAH FOR V16N68 DISPLAY
36 STOVL
37
38 ZEROVECS
39 UNFC/2 # INITIALIZE TRIM VELOCITY CORRECTION TERM
40 HI6ZEROS
41
42 STORE TTF/8
43
44 IGNALoop DLOAD
45
46 TAT
47 STOVL PIPTIME1
48 RATT1
49
50 VSL4 MXV
51 REFSMMAT
52
53 STCALL R
54
55 MUNGRAV
56 GDT/2
57 ?GUIDSUB # WHICH DELIVERS N PASSES OF GUIDANCE
```

# DDUMCALC IS PROGRAMMED AS FOLLOWS:-

```
#
2
(RIGNZ - RGU)/16 + 16(RGU)KIGNY/B8 + (RGU - RIGNX)KIGNX/B4 + (ABVAL(VGU) - VIGN)KIGNV/B4
2 1 0
DDUM = -----
10
2 (VGU - 16 VGU KIGNX/B4)
2 0
```

# DISCONNECTED FROM THEIR RESPECTIVE VARIABLES  
# THE NUMERATOR IS SCALED IN METERS AT 2(28). THE DENOMINATOR IS A VELOCITY IN UNITS OF 2(10) M/CS.  
# THE QUOTIENT IS THUS A TIME IN UNITS OF 2(18) CENTISECONDS. THE FINAL SHIFT RESCALES TO UNITS OF 2(28) CS.  
# THERE IS NO DAMPING FACTOR. THE CONSTANTS KIGNX/B4, KIGNY/B8 AND KIGNV/B4 ARE ALL NEGATIVE IN SIGN.

|          |                   |                                         |                                                           |
|----------|-------------------|-----------------------------------------|-----------------------------------------------------------|
| DDUMCALC | TS<br>TC<br>DLOAD | NIGNLOOP<br>INTPRET<br>DMPR             | # FORM DENOMINATOR FIRST                                  |
|          |                   | VGU<br>KIGNX/B4<br>BDSU                 |                                                           |
|          | SL4R              |                                         |                                                           |
|          | PDDL              | VGU +4<br>DSU<br>RIGNZ                  |                                                           |
|          |                   | RGU +4<br>PDDL<br>RGU +2                |                                                           |
|          | SR4R              |                                         |                                                           |
|          | DSQ               | DMPR<br>KIGNY/B8                        |                                                           |
|          | SL4R              | PDDL                                    |                                                           |
|          |                   | RGU<br>DMPR<br>RIGNX                    |                                                           |
|          | DSU               |                                         |                                                           |
|          |                   | KIGNX/B4<br>ABVAL<br>VGU                |                                                           |
|          | PDVL              |                                         |                                                           |
|          |                   | DMPR<br>VIGN<br>KIGNV/B4                |                                                           |
|          | DSU               |                                         |                                                           |
|          | DAD<br>DAD<br>SRR | DAD<br>DDV                              |                                                           |
|          |                   | 10D                                     |                                                           |
|          | PUSH              | DAD                                     |                                                           |
|          | STODL<br>ABS      | PIPTIME1<br>TDEC1<br>DSU                | # STORE NEW GUESS FOR NEXT INTEGRATION                    |
|          | BMN               | DDUMCRIT<br>CALL<br>DDUMGOOD            |                                                           |
|          | SET               | INTSTALL<br>SET<br>INTYPFLG<br>MOONFLAG |                                                           |
|          | DLOAD             |                                         |                                                           |
|          |                   | PIPTIME1                                |                                                           |
|          | STOVL             | TET<br>RATT1                            | # HOPEFULLY ?GUIDSUB DID NOT<br># CLOBBER RATT1 AND VATT1 |

|    |          |        |          |                                          |
|----|----------|--------|----------|------------------------------------------|
| 1  |          |        |          |                                          |
| 2  |          | STOVL  | RCV      |                                          |
| 3  |          |        | VATT1    |                                          |
| 4  |          | STCALL | VCV      |                                          |
| 5  |          |        | INTEGRVS |                                          |
| 6  |          | GOTO   |          |                                          |
| 7  |          |        | IGNALOOP |                                          |
| 8  |          |        |          |                                          |
| 9  | DDUMGOOD | SLOAD  | SR       |                                          |
| 10 |          |        | ZOOMTIME |                                          |
| 11 |          |        | 14D      |                                          |
| 12 |          | BDSU   |          |                                          |
| 13 |          |        | TDEC1    |                                          |
| 14 |          | STOVL  | TIG      | # COMPUTE DISTANCE LANDING SITE WILL BE  |
| 15 |          |        | V        | # OUT OF LM'S ORBITAL PLANE AT IGNITION: |
| 16 |          | VXV    | UNIT     | # SIGN IS + IF LANDING SITE IS TO THE    |
| 17 |          |        | R        | # RIGHT, NORTH; - IF TO THE LEFT, SOUTH. |
| 18 |          | DOT    | SL1      |                                          |
| 19 |          |        | LAND     |                                          |
| 20 | R60INIT  | STOVL  | OUTOFPLN | # INITIALIZATION FOR CALCMANU            |
| 21 |          |        | UNFC/2   |                                          |
| 22 |          | STORE  | R60VSAVE | # STORE UNFC/2 TEMPORARILY IN R60SAVE    |
| 23 |          | EXIT   |          |                                          |
| 24 |          |        |          | # *****                                  |
| 25 |          |        |          |                                          |
| 26 | IGNALGRT | TC     | PHASCHNG | # PREVENT REPEATING IGNALG               |
| 27 |          | OCT    | 04024    |                                          |
| 28 |          |        |          |                                          |
| 29 | ASTNCLOK | CS     | ASTNDEX  |                                          |
| 30 |          | TC     | BANKCALL |                                          |
| 31 |          | CADR   | STCLOK2  |                                          |
| 32 |          | TCF    | ENDOFJOB | # RETURN IN NEW JOB AND IN EBANK FIVE    |
| 33 |          |        |          |                                          |
| 34 | ASTNRET  | TC     | INTPRET  |                                          |
| 35 |          | SSP    | RTB      | # GO PICK UP DISPLAY AT END OF R51:      |
| 36 |          |        | QMAJ     | # "PROCEED" WILL DO A FINE ALIGNMENT     |
| 37 |          | FCADR  | P63SPOT2 | # " ENTER " WILL RETURN TO P63SPOT2      |
| 38 |          |        | R51P63   |                                          |
| 39 | P63SPOT2 | VLOAD  | UNIT     | # INITIALIZE KALCMANU FOR BURN ATTITUDE  |
| 40 |          |        | R60VSAVE |                                          |
| 41 |          | STOVL  | POINTVSM |                                          |
| 42 |          |        | UNITX    |                                          |
| 43 |          | STORE  | SCAXIS   |                                          |
| 44 |          | EXIT   |          |                                          |
| 45 |          |        |          |                                          |
| 46 |          | CAF    | EBANK7   |                                          |
| 47 |          | TS     | EBANK    |                                          |
| 48 |          |        |          |                                          |
| 49 |          | INHINT |          |                                          |
| 50 |          | TC     | IBNKCALL |                                          |
| 51 |          | CADR   | PFLITEDB |                                          |
| 52 |          |        |          |                                          |
| 53 |          |        |          |                                          |
| 54 |          |        |          |                                          |
| 55 |          |        |          |                                          |
| 56 |          |        |          |                                          |
| 57 |          |        |          |                                          |
| 58 |          |        |          |                                          |
| 59 |          |        |          |                                          |
| 60 |          |        |          |                                          |

|    |          |                                |              |                                           |    |
|----|----------|--------------------------------|--------------|-------------------------------------------|----|
| 1  | RELINT   |                                |              |                                           | 1  |
| 2  |          | TC                             | BANKCALL     |                                           | 2  |
| 3  |          | CADR                           | R60LEM       |                                           | 3  |
| 4  |          |                                |              |                                           | 4  |
| 5  |          | TC                             | PHASCHNG     | # PREVENT RECALLING R60                   | 5  |
| 6  |          | OCT                            | 04024        |                                           | 6  |
| 7  |          |                                |              |                                           | 7  |
| 8  |          |                                |              |                                           | 8  |
| 9  |          |                                |              |                                           | 9  |
| 10 | P63SPOT3 | CA                             | BIT6         | # IS THE LR ANTENNA IN POSITION 1 YET     | 10 |
| 11 |          | EXTEND                         |              |                                           | 11 |
| 12 |          | RAND                           | CHAN33       |                                           | 12 |
| 13 |          | EXTEND                         |              |                                           | 13 |
| 14 |          | BZF                            | P63SPOT4     | # BRANCH IF ANTENNA ALREADY IN POSITION 1 | 14 |
| 15 |          |                                |              |                                           | 15 |
| 16 |          | CAF                            | CODE500      | # ASTRONAUT: PLEASE CRANK THE             | 16 |
| 17 |          | TC                             | BANKCALL     | # SILLY THING AROUND                      | 17 |
| 18 |          | CADR                           | GOPERF1      |                                           | 18 |
| 19 |          | TCF                            | GOTOPOOH     | # TERMINATE                               | 19 |
| 20 |          | TCF                            | P63SPOT3     | # PROCEED SEE IF HE'S LYING               | 20 |
| 21 |          |                                |              |                                           | 21 |
| 22 | P63SPOT4 | TC                             | BANKCALL     | # ENTER INITIALIZE LANDING RADAR          | 22 |
| 23 |          | CADR                           | SETPOS1      |                                           | 23 |
| 24 |          |                                |              |                                           | 24 |
| 25 |          | TC                             | POSTJUMP     | # OFF TO SEE THE WIZARD...                | 25 |
| 26 |          | CADR                           | BURNBABY     |                                           | 26 |
| 27 |          |                                |              |                                           | 27 |
| 28 | #        | -----                          |              |                                           | 28 |
| 29 |          |                                |              |                                           | 29 |
| 30 | #        | CONSTANTS FOR P63LM AND IGNALG |              |                                           | 30 |
| 31 |          |                                |              |                                           | 31 |
| 32 | P63ADRES | GENADR                         | P63TABLE     |                                           | 32 |
| 33 |          |                                |              |                                           | 33 |
| 34 | ASTNDEX  | =                              | MD1          | # OCT 25: INDEX FOR CLOKTASK              | 34 |
| 35 |          |                                |              |                                           | 35 |
| 36 | CODE500  | OCT                            | 00500        |                                           | 36 |
| 37 |          |                                |              |                                           | 37 |
| 38 | 99999CON | 2DEC                           | 30479.7 B-24 |                                           | 38 |
| 39 |          |                                |              |                                           | 39 |
| 40 | GUIDDURN | 2DEC                           | +66440       | # GUIDDURN +6.64400314 E+2                | 40 |
| 41 | DDUMCRIT | 2DEC                           | +8 B-28      | # CRITERION FOR IGNALG CONVERGENCE        | 41 |
| 42 |          |                                |              |                                           | 42 |
| 43 |          |                                |              |                                           | 43 |
| 44 |          |                                |              |                                           | 44 |
| 45 |          |                                |              |                                           | 45 |
| 46 |          |                                |              |                                           | 46 |
| 47 |          |                                |              |                                           | 47 |
| 48 |          |                                |              |                                           | 48 |
| 49 |          |                                |              |                                           | 49 |
| 50 |          |                                |              |                                           | 50 |
| 51 |          |                                |              |                                           | 51 |
| 52 |          |                                |              |                                           | 52 |
| 53 |          |                                |              |                                           | 53 |
| 54 |          |                                |              |                                           | 54 |
| 55 |          |                                |              |                                           | 55 |
| 56 |          |                                |              |                                           | 56 |
| 57 |          |                                |              |                                           | 57 |
| 58 |          |                                |              |                                           | 58 |
| 59 |          |                                |              |                                           | 59 |
| 60 |          |                                |              |                                           | 60 |

#

1412THE

```

P68: LANDING CONFIRMATION

 BANK 31
 SETLOC F2DPS*31
 BANK

COUNT* $$/P6567

LANDJUNK TC PHASCHNG
 OCT 04024

 INHINT

 TC BANKCALL # ZERO ATTITUDE ERROR
 CADR ZATTEROR

 TC BANKCALL # SET 5 DEGREE DEADBAND
 CADR SETMAXDB

 TC INTERPRET # TO INTERPRETIVE AS TIME IS NOT CRITICAL
 SET CLEAR
 SURFFLAG
 LETABORT
 SET VLOAD
 APSFLAG

 STODL RN
 ALPHAV
 PIPTIME

 SET CALL
 LUNAFLAG
 LAT-LONG

 SETPD VLOAD # COMPUTE RLS AND STORE IT AWAY
 0
 RN

 VSL2 PDDL
 PIPTIME

 PUSH CALL
 R-TO-RP
 STORE RLS
 EXIT

 CAF V06N43* # ASTRONAUT: NOW LOOK WHERE YOU ENDED UP
 TC BANKCALL
 CADR GOFLASH

 TCF GOTOPOOH # TERMINATE
 TCF +2 # PROCEED
 TCF -5 # RECYCLE

 TC INTERPRET
```



|    |         |        |          |                                      |    |
|----|---------|--------|----------|--------------------------------------|----|
| 1  |         |        |          |                                      | 1  |
| 2  |         | VLOAD  |          | # INITIALIZE GSAV AND (USING REFMF)  | 2  |
| 3  |         |        | UNITX    | # YNBSAV, ZNBSAV AND ATTFLAG FOR P57 | 3  |
| 4  |         | STCALL | GSAV     |                                      | 4  |
| 5  |         |        | REFMF    |                                      | 5  |
| 6  |         | EXIT   |          |                                      | 6  |
| 7  |         |        |          |                                      | 7  |
| 8  |         | TCF    | GOTOP00H | # ASTRONAUT: PLEASE SELECT P57       | 8  |
| 9  |         |        |          |                                      | 9  |
| 10 | V06N43* | VN     | 0643     |                                      | 10 |
| 11 |         |        |          |                                      | 11 |
| 12 |         |        |          |                                      | 12 |
| 13 |         |        |          |                                      | 13 |
| 14 |         |        |          |                                      | 14 |
| 15 |         |        |          |                                      | 15 |
| 16 |         |        |          |                                      | 16 |
| 17 |         |        |          |                                      | 17 |
| 18 |         |        |          |                                      | 18 |
| 19 |         |        |          |                                      | 19 |
| 20 |         |        |          |                                      | 20 |
| 21 |         |        |          |                                      | 21 |
| 22 |         |        |          |                                      | 22 |
| 23 |         |        |          |                                      | 23 |
| 24 |         |        |          |                                      | 24 |
| 25 |         |        |          |                                      | 25 |
| 26 |         |        |          |                                      | 26 |
| 27 |         |        |          |                                      | 27 |
| 28 |         |        |          |                                      | 28 |
| 29 |         |        |          |                                      | 29 |
| 30 |         |        |          |                                      | 30 |
| 31 |         |        |          |                                      | 31 |
| 32 |         |        |          |                                      | 32 |
| 33 |         |        |          |                                      | 33 |
| 34 |         |        |          |                                      | 34 |
| 35 |         |        |          |                                      | 35 |
| 36 |         |        |          |                                      | 36 |
| 37 |         |        |          |                                      | 37 |
| 38 |         |        |          |                                      | 38 |
| 39 |         |        |          |                                      | 39 |
| 40 |         |        |          |                                      | 40 |
| 41 |         |        |          |                                      | 41 |
| 42 |         |        |          |                                      | 42 |
| 43 |         |        |          |                                      | 43 |
| 44 |         |        |          |                                      | 44 |
| 45 |         |        |          |                                      | 45 |
| 46 |         |        |          |                                      | 46 |
| 47 |         |        |          |                                      | 47 |
| 48 |         |        |          |                                      | 48 |
| 49 |         |        |          |                                      | 49 |
| 50 |         |        |          |                                      | 50 |
| 51 |         |        |          |                                      | 51 |
| 52 |         |        |          |                                      | 52 |
| 53 |         |        |          |                                      | 53 |
| 54 |         |        |          |                                      | 54 |
| 55 |         |        |          |                                      | 55 |
| 56 |         |        |          |                                      | 56 |
| 57 |         |        |          |                                      | 57 |
| 58 |         |        |          |                                      | 58 |
| 59 |         |        |          |                                      | 59 |
| 60 |         |        |          |                                      | 60 |

```
1
2 BANK 31
3 SETLOC FTHROT
4
5 BANK
6 EBANK= PIF
7 COUNT* $$/THROT
8
9 # * * * * *
10 # HERE FC, DESIRED THRUST, AND FP, PRESENT THRUST, UNWEIGHTED, ARE COMPUTED.
11
12 THROTTLE CA ABDELV # COMPUTE PRESENT ACCELERATION IN UNITS OF
13 EXTEND # 2(-4) M/CS/CS, SAVING SERVICER TROUBLE
14
15 +3 MP /AF/CNST
16 EXTEND
17 AFDUMP QXCH RTNHOLD
18 TC MASSMULT
19 DXCH FP # FP = PRESENT THRUST
20 EXTEND
21
22 DCA /AFC/
23 TC MASSMULT
24 TS FC # FC = THRUST DESIRED BY GUIDANCE
25 DXCH FCODD # FCODD = WHAT IT IS GOING TO GET
26
27 # IF IT HAS BEEN LESS THAN 3 SECONDS SINCE THE LAST THROTTLING, AUGMENT FP USING THE FWEIGHT CALCULATED THEN.
28
29 CS TTHROT # THIS CODING ASSUMES A FLATOUT WITHIN
30 AD TIME1 # 80 SECONDS BEFORE FIRST THROTTLE CALL
31 MASK POSMAX
32 COM
33 AD 3SECS
34
35 EXTEND
36 BZMF WHERETO # BRANCH IF (TIME1-TTHROT +1) > 3 SECONDS
37 EXTEND
38 DCA FWEIGHT
39 DAS FP
40
41 # THIS LOGIC DETERMINES THE THROTTLING IN THE REGION 10% - 94%. THE MANUAL THROTTLE, NOMINALLY SET AT
42 # MINIMUM BY ASTRONAUT OR MISSION CONTROL PROGRAMS, PROVIDES THE LOWER BOUND. A STOP IN THE THROTTLE HARDWARE
43 # PROVIDES THE UPPER.
44
45 WHERETO CA EBANK5 # INITIALIZE L*WCR*T AND H*GHCR*T FROM
46 TS EBANK # PAD LOADED ERASABLES IN W-MATRIX
```



```
1
2 EBANK= LOWCRIT
3 EXTEND
4 DCA LOWCRIT
5 DXCH L*WCR*T
6 CA EBANK7
7
8 TS EBANK
9 EBANK= PIF
10 CS ZERO # INITIALIZE PIFPSET
11
12 TS PIFPSET
13 CS H*GHCR*T
14 AD FCOLD
15
16 EXTEND
17 BZMF LOWFCOLD # BRANCH IF FCOLD < OR = HIGHCRIT
18 CS L*WCR*T
19
20 AD FCODD
21 EXTEND
22 BZMF FCOMPSET # BRANCH IF FC < OR = LOWCRIT
23 CA FP # SEE NOTE 1
24 TCF FLATOUT1
25
26 FCOMPSET CS FMAXODD # SEE NOTE 2
27 AD FP
28 TCF FLATOUT2
29
30 LOWFCOLD CS H*GHCR*T
31 AD FCODD
32
33 EXTEND
34 BZMF DOPIF # BRANCH IF FC < OR = HIGHCRIT
35
36 FLATOUT1 CA FMAXPOS # NO: THROTTLE-UP
37 DXCH FCODD
38 CA FEXTRA
39
40 FLATOUT2 TS PIFPSET
41
42 # NOTE 1 FC IS SET EQUAL TO FP SO PIF WILL BE ZERO. THIS IS DESIRABLE
43 # AS THERE IS ACTUALLY NO THROTTLE CHANGE.
44 #
45 # NOTE 2 HERE, SINCE WE ARE ABOUT TO RETURN TO THE THROTTLEABLE REGION
46 # (BELOW 55%) THE QUANTITY -(FMAXODD - FP) IS COMPUTED AND PUT
47 # INTO PIFPSET TO COMPENSATE FOR THE DIFFERENCE BETWEEN THE
48 # NUMBER OF BITS CORRESPONDING TO FULL THROTTLE (FMAXODD) AND THE
49 # NUMBER CORRESPONDING TO ACTUAL THRUST (FP). THUS THE TOTAL
50 # THROTTLE COMMAND PIF = FC - FP - (FMAXODD - FP) = FC - FMAXODD.
51
52 DOPIF TC FASTCHNG
53 EXTEND
54 DCA FCODD
55
56 TS FCOLD
57 DXCH PIF
58 EXTEND
```

```
1
2 DCS FP
3 DAS PIF # PIF = FC - FP, NEVER EQUALS +0
4
5 DOIT CA PIF
6 AD PIFPSET # ADD IN PIFPSET, WITHOUT CHANGING PIF
7 TS PSEUDO55
8 TS THRUST
9 CAF BIT4
10 EXTEND
11 WOR CHAN14
12 CA TIME1
13 TS TTHROT
14
15 # SINCE /AF/ IS NOT AN INSTANTANEOUS ACCELERATION, BUT RATHER AN "AVERAGE" OF THE ACCELERATION LEVELS DURING
16 # THE PRECEEDING PIPA INTERVAL, AND SINCE FP IS COMPUTED DIRECTLY FROM /AF/, FP IN ORDER TO CORRESPOND TO THE
17 # ACTUAL THRUST LEVEL AT THE END OF THE INTERVAL MUST BE WEIGHTED BY
18 #
19 # PIF(PPROCESS + TL) PIF /PIF/
20 # FWEIGHT = ----- + -----
21 # PGUID 2 PGUID FRATE
22 #
23 # WHERE PPROCESS IS THE TIME BETWEEN PIPA READING AND THE START OF THROTTLING, PGUID IS THE GUIDANCE PERIOD, AND
24 # FRATE IS THE THROTTLING RATE (32 UNITS PER CENTISECOND). PGUID IS EITHER 1 OR 2 SECONDS. THE "TL" IN THE
25 # FIRST TERM REPRESENTS THE ENGINE'S RESPONSE LAG. HERE FWEIGHT IS COMPUTED FOR USE NEXT PASS.
26
27 CA THISTPIP +1 # INITIALIZE FWEIGHT COMP AS IF FOR P66
28 TS BUF
29
30 CS MODREG # ARE WE IN FACT IN P66?
31 AD DEC66
32 EXTEND
33 BZF FWCOMP # YES
34
35 CA PIPTIME +1 # NO: INITIALIZE FOR TWO SECOND PERIOD
36 TS BUF
37 CAF 4SECS
38 TCF FWCOMP +1
39
40 FWCOMP CAF 2SECS
41 +1 TS Q
42 EXTEND
43 MP BIT6
44 LXCH BUF +1
45 CS BUF # TIME OF LAST PIPA READING.
46 AD TIME1
47 AD THROTLAG # COMPENSATE FOR ENGINE RESPONSE LAG
48 MASK LOW8 # MAKE SURE SMALL AND POSITIVE
49 ZL
50 EXTEND
51
52
53
54
55
56
57
58
59
60
```

# THROTTLE\_CONTROL\_ROUTINES

|    |                                                                                                         |        |                                               |    |
|----|---------------------------------------------------------------------------------------------------------|--------|-----------------------------------------------|----|
| 1  |                                                                                                         |        |                                               | 1  |
| 2  |                                                                                                         | DV     | Q                                             | 2  |
| 3  |                                                                                                         | EXTEND |                                               | 3  |
| 4  |                                                                                                         | MP     | PIF                                           | 4  |
| 5  |                                                                                                         | DOUBLE |                                               | 5  |
| 6  |                                                                                                         | DXCH   | FWEIGHT                                       | 6  |
| 7  |                                                                                                         | CCS    | PIF                                           | 7  |
| 8  |                                                                                                         | AD     | ONE                                           | 8  |
| 9  |                                                                                                         | TCF    | +2                                            | 9  |
| 10 |                                                                                                         | AD     | ONE                                           | 10 |
| 11 |                                                                                                         | EXTEND |                                               | 11 |
| 12 |                                                                                                         | MP     | PIF                                           | 12 |
| 13 |                                                                                                         | EXTEND |                                               | 13 |
| 14 |                                                                                                         | DV     | BUF +1                                        | 14 |
| 15 |                                                                                                         | ZL     |                                               | 15 |
| 16 |                                                                                                         | DAS    | FWEIGHT                                       | 16 |
| 17 |                                                                                                         |        |                                               | 17 |
| 18 | THDUMP                                                                                                  | TC     | RTNHOLD                                       | 18 |
| 19 |                                                                                                         |        |                                               | 19 |
| 20 | # FLATOUT THROTTLES UP THE DESCENT ENGINE, AND IS CALLED AS A BASIC SUBROUTINE.                         |        |                                               | 20 |
| 21 |                                                                                                         |        |                                               | 21 |
| 22 | FLATOUT                                                                                                 | CAF    | BIT13 # 4096 PULSES                           | 22 |
| 23 | WHATOUT                                                                                                 | TS     | PIFPSET # USE PIFPSET SO FWEIGHT WILL BE ZERO | 23 |
| 24 |                                                                                                         | CS     | ZERO                                          | 24 |
| 25 |                                                                                                         | TS     | FCOLD                                         | 25 |
| 26 |                                                                                                         | TS     | PIF                                           | 26 |
| 27 |                                                                                                         | EXTEND |                                               | 27 |
| 28 |                                                                                                         | QXCH   | RTNHOLD                                       | 28 |
| 29 |                                                                                                         | TCF    | DOIT                                          | 29 |
| 30 |                                                                                                         |        |                                               | 30 |
| 31 | # MASSMULT SCALES ACCELERATION, ARRIVING IN A AND L IN UNITS OF 2(-4) M/CS/CS, TO FORCE IN PULSE UNITS. |        |                                               | 31 |
| 32 |                                                                                                         |        |                                               | 32 |
| 33 | MASSMULT                                                                                                | EXTEND |                                               | 33 |
| 34 |                                                                                                         | QXCH   | BUF                                           | 34 |
| 35 |                                                                                                         | DXCH   | MPAC                                          | 35 |
| 36 |                                                                                                         | TC     | DMP                                           | 36 |
| 37 |                                                                                                         | ADRES  | MASS                                          | 37 |
| 38 |                                                                                                         | TC     | DMP # LEAVES PROPERLY SCALED FORCE IN MPAC    | 38 |
| 39 |                                                                                                         | ADRES  | SCALEFAC                                      | 39 |
| 40 |                                                                                                         | TC     | TPAGREE                                       | 40 |
| 41 |                                                                                                         | CA     | MPAC                                          | 41 |
| 42 |                                                                                                         | EXTEND |                                               | 42 |
| 43 |                                                                                                         | BZF    | +3                                            | 43 |
| 44 |                                                                                                         | CAF    | POSMAX                                        | 44 |
| 45 |                                                                                                         | TC     | BUF                                           | 45 |
| 46 |                                                                                                         | DXCH   | MPAC +1                                       | 46 |
| 47 |                                                                                                         | TC     | BUF                                           | 47 |
| 48 |                                                                                                         |        |                                               | 48 |
| 49 |                                                                                                         |        |                                               | 49 |
| 50 |                                                                                                         |        |                                               | 50 |
| 51 |                                                                                                         |        |                                               | 51 |
| 52 |                                                                                                         |        |                                               | 52 |
| 53 |                                                                                                         |        |                                               | 53 |
| 54 |                                                                                                         |        |                                               | 54 |
| 55 |                                                                                                         |        |                                               | 55 |
| 56 |                                                                                                         |        |                                               | 56 |
| 57 |                                                                                                         |        |                                               | 57 |
| 58 |                                                                                                         |        |                                               | 58 |
| 59 |                                                                                                         |        |                                               | 59 |
| 60 |                                                                                                         |        |                                               | 60 |



|    |                                       |    |
|----|---------------------------------------|----|
| 1  | # CONSTANTS:-                         | 1  |
| 2  |                                       | 2  |
| 3  |                                       | 3  |
| 4  | FEXTRA = BIT13 # FEXT +5.13309020E+ 4 | 4  |
| 5  |                                       | 5  |
| 6  | /AF/CNST DEC .13107                   | 6  |
| 7  |                                       | 7  |
| 8  | # * * * * *                           | 8  |
| 9  |                                       | 9  |
| 10 |                                       | 10 |
| 11 |                                       | 11 |
| 12 |                                       | 12 |
| 13 |                                       | 13 |
| 14 |                                       | 14 |
| 15 |                                       | 15 |
| 16 |                                       | 16 |
| 17 |                                       | 17 |
| 18 |                                       | 18 |
| 19 |                                       | 19 |
| 20 |                                       | 20 |
| 21 |                                       | 21 |
| 22 |                                       | 22 |
| 23 |                                       | 23 |
| 24 |                                       | 24 |
| 25 |                                       | 25 |
| 26 |                                       | 26 |
| 27 |                                       | 27 |
| 28 |                                       | 28 |
| 29 |                                       | 29 |
| 30 |                                       | 30 |
| 31 |                                       | 31 |
| 32 |                                       | 32 |
| 33 |                                       | 33 |
| 34 |                                       | 34 |
| 35 |                                       | 35 |
| 36 |                                       | 36 |
| 37 |                                       | 37 |
| 38 |                                       | 38 |
| 39 |                                       | 39 |
| 40 |                                       | 40 |
| 41 |                                       | 41 |
| 42 |                                       | 42 |
| 43 |                                       | 43 |
| 44 |                                       | 44 |
| 45 |                                       | 45 |
| 46 |                                       | 46 |
| 47 |                                       | 47 |
| 48 |                                       | 48 |
| 49 |                                       | 49 |
| 50 |                                       | 50 |
| 51 |                                       | 51 |
| 52 |                                       | 52 |
| 53 |                                       | 53 |
| 54 |                                       | 54 |
| 55 |                                       | 55 |
| 56 |                                       | 56 |
| 57 |                                       | 57 |
| 58 |                                       | 58 |
| 59 |                                       | 59 |
| 60 |                                       | 60 |

EBANK= E2DPS  
COUNT\* \$\$/F2DPS

# \*\*\*\*\*

# LUNAR LANDING FLIGHT SEQUENCE TABLES

# \*\*\*\*\*

# FLIGHT SEQUENCE TABLES ARE ARRANGED BY FUNCTION. THEY ARE REFERENCED USING AS AN INDEX THE REGISTER WCHPHASE:

# WCHPHASE = -1 ---&gt; IGNALG

# WCHPHASE = 0 ---&gt; BRAKQUAD

# WCHPHASE = 1 ---&gt; APPRQUAD

# WCHPHASE = 2 ---&gt; VERTICAL

#\*\*\*\*\*

# ROUTINES FOR STARTING NEW GUIDANCE PHASES:

|          |     |          |            |
|----------|-----|----------|------------|
| NEWPHASE | TCF | TTFINCR  | # IGNALG   |
|          | TCF | TTFINCR  | # BRAKQUAD |
|          | TCF | STARTP64 | # APPRQUAD |
|          | TCF | P65START | # VERTICAL |

# PRE-GUIDANCE COMPUTATIONS:

|          |     |          |            |
|----------|-----|----------|------------|
| PREGUIDE | TCF | CALCRGVG | # IGNALG   |
|          | TCF | RGVGCALC | # BRAKQUAD |
|          | TCF | REDESIG  | # APPRQUAD |
|          | TCF | RGVGCALC | # VERTICAL |

# GUIDANCE EQUATIONS:

|          |     |          |            |
|----------|-----|----------|------------|
| WHATGUID | TCF | TTF/8CL  | # IGNALG   |
|          | TCF | TTF/8CL  | # BRAKQUAD |
|          | TCF | TTF/8CL  | # APPRQUAD |
|          | TCF | VERTGUID | # VERTICAL |

# POST GUIDANCE EQUATION COMPUTATIONS:

|          |     |        |            |
|----------|-----|--------|------------|
| AFTRGUID | TCF | CGCALC | # IGNALG   |
|          | TCF | CGCALC | # BRAKQUAD |
|          | TCF | CGCALC | # APPRQUAD |
|          | TCF | STEER? | # VERTICAL |

# WINDOW VECTOR COMPUTATIONS:

|          |     |        |            |
|----------|-----|--------|------------|
| WHATEXIT | TCF | EXGSUB | # IGNALG   |
|          | TCF | EXBRAK | # BRAKQUAD |
|          | TCF | EXNORM | # APPRQUAD |

# DISPLAY ROUTINES:

|          |     |          |            |
|----------|-----|----------|------------|
| WHATDISP | TCF | P63DISPS | # BRAKQUAD |
|          | TCF | P64DISPS | # APPRQUAD |
|          | TCF | VERTDISP | # VERTICAL |

# ALARM ROUTINE FOR TTF COMPUTATION:

|         |     |         |            |
|---------|-----|---------|------------|
| WHATALM | TCF | 1406P00 | # IGNALG   |
|         | TCF | 1406ALM | # BRAKQUAD |
|         | TCF | 1406ALM | # APPRQUAD |

# INDICES FOR REFERENCING TARGET PARAMETERS

|           |     |    |            |
|-----------|-----|----|------------|
| TARGETDEX | OCT | 0  | # IGNALG   |
|           | OCT | 0  | # BRAKQUAD |
|           | OCT | 34 | # APPRQUAD |

\*\*\*\*\*  
# ENTRY POINTS: ?GUIDSUB FOR THE IGNITION ALGORITHM, LUNLAND FOR SERVOUT  
\*\*\*\*\*

# IGNITION ALGORITHM ENTRY: DELIVERS N PASSES OF QUADRATIC GUIDANCE

|          |      |             |                                     |
|----------|------|-------------|-------------------------------------|
| ?GUIDSUB | EXIT |             |                                     |
|          | CAF  | TWO         | # N = 3                             |
|          | TS   | NGUIDSUB    |                                     |
|          | TCF  | GUILDRET +2 |                                     |
| GUIDSUB  | TS   | NGUIDSUB    | # ON SUCCEEDING PASSES SKIP TTFINCR |
|          | TCF  | CALCRGVG    |                                     |

# NORMAL ENTRY: CONTROL COMES HERE FROM SERVOUT

|         |     |          |                                          |
|---------|-----|----------|------------------------------------------|
| LUNLAND | TC  | PHASCHNG |                                          |
|         | OCT | 00035    | # GROUP 5: RETAIN ONLY PIPA TASK         |
|         | TC  | PHASCHNG |                                          |
|         | OCT | 05023    | # GROUP 3: PROTECT GUIDANCE WITH PRIO 21 |
|         | OCT | 21000    | # JUST HIGHER THAN SERVICER'S PRIORITY   |

\*\*\*\*\*

# GUILDENSTERN: AUTO-MODES MONITOR (R13)

\*\*\*\*\*

COUNT\* \$\$/R13

# HERE IS THE PHILOSOPHY OF GUILDENSTERN: ON EVERY APPEARANCE OR DISAPPEARANCE OF THE MANUAL THROTTLE

# DISCRETE TO SELECT P67 OR P66 RESPECTIVELY: ON EVERY APPEARANCE OF THE ATTITUDE-HOLD DISCRETE TO SELECT P66

# UNLESS THE CURRENT PROGRAM IS P67 IN WHICH CASE THERE IS NO CHANGE

GUILDEN EXTEND # IS UN-AUTO-THROTTLE DISCRETE PRESENT?

# STERN # RSB 2009: NOT ORIGINALLY A COMMENT.

READ CHAN30

MASK BIT5

CCS A

TCF STARTP67 # YES

P67NOW? TC CHECKMM # NO: ARE WE IN P67 NOW?

DEC 67

TCF STABL? # NO

STARTP66 TC FASTCHNG # YES

TC NEWMODEX

DEC66 DEC 66

EXTEND

DCA HDOTDISP # SET DESIRED ALTITUDE RATE = CURRENT

DXCH VDGVERT # ALTITUDE RATE.

STRTP66A TC INTPRET

SLOAD PUSH

PBIASZ

SLOAD PUSH

PBIASY

SLOAD VDEF

PBIASX

VXSC SET

BIASFACT

RODFLAG

STOVL VBIAS

TEMX

VCOMP

STOVL OLDPIPAX

ZEROVECS

STODL DELVROD

RODSSCALE

STODL RODSCAL1

PIPTIME

STORE LASTTPIP

EXIT

CAF ZERO

TS FCOLD

TS FWEIGHT

TS FWEIGHT +1

VRTSTART TS WCHVERT

|    |          |                              |            |                                           |
|----|----------|------------------------------|------------|-------------------------------------------|
| 1  |          |                              |            |                                           |
| 2  |          | CAF                          | TWO        | # WCHPHASE = 2 ---> VERTICAL: P65,P66,P67 |
| 3  |          | TS                           | WCHPHOLD   |                                           |
| 4  |          | TS                           | WCHPHASE   |                                           |
| 5  |          | TC                           | BANKCALL   | # TEMPORARY, I HOPE HOPE HOPE             |
| 6  |          | CADR                         | STOPRATE   | # TEMPORARY, I HOPE HOPE HOPE             |
| 7  |          | TC                           | DOWNFLAG   | # PERMIT X-AXIS OVERRIDE                  |
| 8  |          | ADRES                        | XOVINFLG   |                                           |
| 9  |          | TC                           | DOWNFLAG   |                                           |
| 10 |          | ADRES                        | REDFLAG    |                                           |
| 11 |          | TCF                          | VERTGUID   |                                           |
| 12 |          |                              |            |                                           |
| 13 | STARTP67 | TC                           | NEWMODEX   | # NO HARM IN "STARTING" P67 OVER AND OVER |
| 14 |          | DEC                          | 67         | # SO NO NEED FOR A FASTCHNG AND NO NEED   |
| 15 |          | CAF                          | ZERO       | # TO SEE IF ALREADY IN P67.               |
| 16 |          | TS                           | RODCOUNT   |                                           |
| 17 |          | CAF                          | TEN        |                                           |
| 18 |          | TCF                          | VRTSTART   |                                           |
| 19 |          |                              |            |                                           |
| 20 | STABL?   | CAF                          | BIT13      | # IS UN-ATTITUDE-HOLD DISCRETE PRESENT?   |
| 21 |          | EXTEND                       |            |                                           |
| 22 |          | RAND                         | CHAN31     |                                           |
| 23 |          | CCS                          | A          |                                           |
| 24 |          | TCF                          | GUILDRET   | # YES ALL'S WELL                          |
| 25 |          |                              |            |                                           |
| 26 | P66NOW?  | CS                           | MODREG     |                                           |
| 27 |          | AD                           | DEC66      |                                           |
| 28 |          | EXTEND                       |            |                                           |
| 29 |          | BZF                          | RESTART?   |                                           |
| 30 |          |                              |            |                                           |
| 31 |          | CA                           | RODCOUNT   | # NO. HAS THE ROD SWITCH BEEN "CLICKED"?  |
| 32 |          | EXTEND                       |            |                                           |
| 33 |          | BZF                          | GUILDRET   | # NO. CONTINUE WITH AUTOMATIC LANDING     |
| 34 |          | TCF                          | STARTP66   | # YES. SWITCH INTO THE ROD MODE.          |
| 35 |          |                              |            |                                           |
| 36 | RESTART? | CA                           | FLAGWRD1   | # HAS THERE BEEN A RESTART?               |
| 37 |          | MASK                         | RODFLBIT   |                                           |
| 38 |          | EXTEND                       |            |                                           |
| 39 |          | BZF                          | STRTP66A   | # YES. REINITIALIZE BUT LEAVE VDGVERT AS  |
| 40 |          |                              |            | # IS.                                     |
| 41 |          |                              |            |                                           |
| 42 |          | TCF                          | VERTGUID   | # NO: CONTINUE WITH R.O.D.                |
| 43 |          |                              |            |                                           |
| 44 | #        | *****                        |            |                                           |
| 45 | #        | INITIALIZATION FOR THIS PASS |            |                                           |
| 46 | #        | *****                        |            |                                           |
| 47 |          |                              |            |                                           |
| 48 |          | COUNT*                       | \$\$/F2DPS |                                           |
| 49 |          |                              |            |                                           |
| 50 | GUILDRET | CAF                          | ZERO       |                                           |
| 51 |          | TS                           | RODCOUNT   |                                           |
| 52 |          |                              |            |                                           |
| 53 |          |                              |            |                                           |
| 54 |          |                              |            |                                           |
| 55 |          |                              |            |                                           |
| 56 |          |                              |            |                                           |
| 57 |          |                              |            |                                           |
| 58 |          |                              |            |                                           |
| 59 |          |                              |            |                                           |
| 60 |          |                              |            |                                           |



```

INCREMENT TTF/8, UPDATE LAND FOR LUNAR ROTATION, DO OTHER USEFUL THINGS

#
TTFINCR COMPUTATIONS ARE AS FOLLOWS --

```

# LUNAR\_LANDING\_GUIDANCE\_EQUATIONS

|    |         |                                                                                                                                                                                                                                        |             |                                          |    |
|----|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|------------------------------------------|----|
| 1  |         |                                                                                                                                                                                                                                        |             |                                          | 1  |
| 2  | #       | TTF/8 UPDATED FOR TIME SINCE LAST PASS:                                                                                                                                                                                                |             |                                          | 2  |
| 3  | #       | TTF/8 = TTF/8 + (TPIP - TPIPOLD)/8                                                                                                                                                                                                     |             |                                          | 3  |
| 4  | #       | LANDING SITE VECTOR UPDATED FOR LUNAR ROTATION:                                                                                                                                                                                        |             |                                          | 4  |
| 5  | #       |                                                                                                                                                                                                                                        |             |                                          | 5  |
| 6  | #       | $\bar{\bar{L}}\bar{\bar{A}}\bar{\bar{N}}\bar{\bar{D}} = /LAND/ \text{ UNIT}(\bar{\bar{L}}\bar{\bar{A}}\bar{\bar{N}}\bar{\bar{D}} - \bar{\bar{L}}\bar{\bar{A}}\bar{\bar{N}}\bar{\bar{D}}(TPIP - TPIPOLD) * \bar{\bar{W}}\bar{\bar{M}})$ |             |                                          | 6  |
| 7  | #       | SLANT RANGE TO LANDING SITE, FOR DISPLAY:                                                                                                                                                                                              |             |                                          | 7  |
| 8  | #       |                                                                                                                                                                                                                                        |             |                                          | 8  |
| 9  | #       | RANGEDSP = ABVAL( $\bar{\bar{L}}\bar{\bar{A}}\bar{\bar{N}}\bar{\bar{D}}$ - $\bar{R}$ )                                                                                                                                                 |             |                                          | 9  |
| 10 |         |                                                                                                                                                                                                                                        |             |                                          | 10 |
| 11 | TTFINCR | TC                                                                                                                                                                                                                                     | INTPRET     |                                          | 11 |
| 12 |         | DLOAD                                                                                                                                                                                                                                  | DSU         |                                          | 12 |
| 13 |         |                                                                                                                                                                                                                                        | TPIP        |                                          | 13 |
| 14 |         |                                                                                                                                                                                                                                        | TPIPOLD     |                                          | 14 |
| 15 |         | SLR                                                                                                                                                                                                                                    | PUSH        | # SHIFT SCALES DELTA TIME TO 2(17) CSECS | 15 |
| 16 |         |                                                                                                                                                                                                                                        | IID         |                                          | 16 |
| 17 |         | VXSC                                                                                                                                                                                                                                   | VXV         |                                          | 17 |
| 18 |         |                                                                                                                                                                                                                                        | LAND        |                                          | 18 |
| 19 |         |                                                                                                                                                                                                                                        | WM          |                                          | 19 |
| 20 |         | BVSU                                                                                                                                                                                                                                   | RTB         |                                          | 20 |
| 21 |         |                                                                                                                                                                                                                                        | LAND        |                                          | 21 |
| 22 |         |                                                                                                                                                                                                                                        | NORMUNIT    |                                          | 22 |
| 23 |         | VXSC                                                                                                                                                                                                                                   | VSL1        |                                          | 23 |
| 24 |         |                                                                                                                                                                                                                                        | /LAND/      |                                          | 24 |
| 25 |         | STODL                                                                                                                                                                                                                                  | LANDTEMP    |                                          | 25 |
| 26 |         | EXIT                                                                                                                                                                                                                                   |             |                                          | 26 |
| 27 |         |                                                                                                                                                                                                                                        |             |                                          | 27 |
| 28 |         | DXCH                                                                                                                                                                                                                                   | MPAC        |                                          | 28 |
| 29 |         | DAS                                                                                                                                                                                                                                    | TTF/8TMP    | # NOW HAVE INCREMENTED TTF/8 IN TTF/8TMP | 29 |
| 30 |         |                                                                                                                                                                                                                                        |             |                                          | 30 |
| 31 |         | TC                                                                                                                                                                                                                                     | FASTCHNG    |                                          | 31 |
| 32 |         |                                                                                                                                                                                                                                        |             |                                          | 32 |
| 33 |         | EXTEND                                                                                                                                                                                                                                 |             |                                          | 33 |
| 34 |         | DCA                                                                                                                                                                                                                                    | TTF/8TMP    |                                          | 34 |
| 35 |         | DXCH                                                                                                                                                                                                                                   | TTF/8       |                                          | 35 |
| 36 |         |                                                                                                                                                                                                                                        |             |                                          | 36 |
| 37 |         | EXTEND                                                                                                                                                                                                                                 |             |                                          | 37 |
| 38 |         | DCA                                                                                                                                                                                                                                    | LANDTEMP    |                                          | 38 |
| 39 |         | DXCH                                                                                                                                                                                                                                   | LAND        |                                          | 39 |
| 40 |         | EXTEND                                                                                                                                                                                                                                 |             |                                          | 40 |
| 41 |         | DCA                                                                                                                                                                                                                                    | LANDTEMP +2 |                                          | 41 |
| 42 |         | DXCH                                                                                                                                                                                                                                   | LAND +2     |                                          | 42 |
| 43 |         | EXTEND                                                                                                                                                                                                                                 |             |                                          | 43 |
| 44 |         | DCA                                                                                                                                                                                                                                    | LANDTEMP +4 |                                          | 44 |
| 45 |         | DXCH                                                                                                                                                                                                                                   | LAND +4     |                                          | 45 |
| 46 |         |                                                                                                                                                                                                                                        |             |                                          | 46 |
| 47 |         |                                                                                                                                                                                                                                        |             |                                          | 47 |
| 48 |         |                                                                                                                                                                                                                                        |             |                                          | 48 |
| 49 |         |                                                                                                                                                                                                                                        |             |                                          | 49 |
| 50 |         |                                                                                                                                                                                                                                        |             |                                          | 50 |
| 51 |         |                                                                                                                                                                                                                                        |             |                                          | 51 |
| 52 |         |                                                                                                                                                                                                                                        |             |                                          | 52 |
| 53 |         |                                                                                                                                                                                                                                        |             |                                          | 53 |
| 54 |         |                                                                                                                                                                                                                                        |             |                                          | 54 |
| 55 |         |                                                                                                                                                                                                                                        |             |                                          | 55 |
| 56 |         |                                                                                                                                                                                                                                        |             |                                          | 56 |
| 57 |         |                                                                                                                                                                                                                                        |             |                                          | 57 |
| 58 |         |                                                                                                                                                                                                                                        |             |                                          | 58 |
| 59 |         |                                                                                                                                                                                                                                        |             |                                          | 59 |
| 60 |         |                                                                                                                                                                                                                                        |             |                                          | 60 |

```
1
2 TC TDISPSET
3 TC FASTCHNG # SINCE REDESIG MAY CHANGE LANDTEMP
4
5 BRSPOT2 INDEX WCHPHASE
6 TCF PREGUIDE
7
8 # *****
9 # LANDING SITE PERTURBATION EQUATIONS
10 # *****
11
12 REDESIG CA FLAGWRD6 # IS REDFLAG SET?
13 MASK REDFLBIT
14 EXTEND
15 BZF RGVGCALC # NO: SKIP REDESIGNATION LOGIC
16
17 CA TREDES # YES: HAS TREDES REACHED ZERO?
18 EXTEND
19 BZF RGVGCALC # YES: SKIP REDESIGNATION LOGIC
20
21 INHINT
22 CA ELINCR1
23 TS ELINCR
24 CA AZINCR1
25 TS AZINCR
26 TC FASTCHNG
27
28 CA ZERO
29 TS ELINCR1
30 TS AZINCR1
31 TS ELINCR +1
32 TS AZINCR +1
33
34 CA FIXLOC # SET PD TO 0
35 TS PUSHLOC
36
37 TC INTPRET
38 VLOAD VSU
39 LAND
40
41 RTB R #
42 PUSH # PUSH DOWN UNIT ($\bar{L}\bar{A}\bar{N}\bar{D} - \bar{R}$)
43 NORMUNIT
44 VXV VSL1
45 YNBPIP #
46 VXSC PDDL # PUSH DOWN - $\text{ELINCR}(\bar{Y}\bar{N}\bar{B} * \text{UNIT}(\bar{L}\bar{A}\bar{N}\bar{D} - \bar{R}))$
47
48 VXSC ELINCR
49 AZINCR
50 VSU
51 VAD YNBPIP # RESULTING VECTOR IS 1/2 REAL SIZE
52 PUSH
53
54
55
56
57
58
59
60
```

```

 DLOAD DSU # MAKE SURE REDESIGNATION IS NOT
 0 # TOO CLOSE TO THE HORIZON.
 BMN DEPRCRIT
 REDES1 DLOAD DSU
 LAND
 R
 DDV VXSC
 VAD 0
 VXSC UNIT
 R
 STORE VSL1
 EXIT /LAND/
 LANDTEMP # LOOKANGL WILL BE COMPUTED AT RGVGCALC
 TC FASTCHNG
 EXTEND
 DCA LANDTEMP
 DXCH LAND
 EXTEND
 DCA LANDTEMP +2
 DXCH LAND +2
 EXTEND
 DCA LANDTEMP +4
 DXCH LAND +4
 TCF RGVGCALC
```

```

COMPUTE STATE IN GUIDANCE COORDINATES

#
RGVGCALC COMPUTATIONS ARE AS FOLLOWS:--
VELOCITY RELATIVE TO THE SURFACE:
#
ANGTERM = $\bar{V} + \bar{R} * \bar{WM}$
STATE IN GUIDANCE COORDINATES:
*
$\bar{RGU} = CG (\bar{R} - \bar{LAND})$
*
$\bar{VGU} = CG (\bar{V} - \bar{WM} * \bar{R})$
```

```
#
HORIZONTAL VELOCITY FOR DISPLAY
#
VHORIZ = 8 ABVAL (0, VG , VG)
2 1
#
DEPRESSION ANGLE FOR DISPLAY:
#
LOOKANGL = ARCSIN(UNIT(R - LAND).XMBPIP)
#
CALCRGVG TC INTPRET # IN IGNALG, COMPUTE V FROM INTEGRATION
 VLOAD MXV # OUTPUT AND TRIM CORRECTION TERM
 VATT1 # COMPUTED LAST PASS AND LEFT IN UNFC/2
 REFSMMAT
 VSR1 VAD
 UNFC/2
 STORE V
 EXIT
#
RGVGCALC TC INTPRET # ENTER HERE TO RECOMPUTE RG AND VG
 VLOAD VXV
 R
 WM
 VAD VSR2 # RESCALE TO UNITS OF 2(9) M/CS
 V
 STORE ANGTERM
 MXV
#
 CG # NO SHIFT SINCE ANGTERM IS DOUBLE SIZED
 STORE VGU
 PDDL VDEF # FORM (0,VG ,VG) IN UNITS OF 2(10) M/CS
 ZEROVECS # 2 1
 ABVAL SL3
 STOVL VHORIZ # VHORIZ FOR DISPLAY DURING P65.
 R
 VSU PUSH # PUSH DOWN R - LAND
 LAND
 MXV VSL1
 CG
 STORE RGU
 ABVAL
 STOVL RANGEDSP
 RTB DOT # NOW IN MPAC IS SINE(LOOKANGL)/4
 NORMUNIT
 XNBPIP
 EXIT
#
 CA FIXLOC # RESET PUSH DOWN POINTER
 TS PUSHLOC
```

```
1
2 CA MPAC # COMPUTE LOOKANGLE ITSELF
3 DOUBLE
4 TC BANKCALL
5 CADR SPARCSIN -1
6 AD 1/2DEG
7 EXTEND
8 MP 180DEGS
9 TS LOOKANGL # LOOKANGL FOR DISPLAY DURING P64
10
11 BRSPOT3 INDEX WCHPHASE
12 TCF WHATGUID
13
14 # *****
15 # TTF/8 COMPUTATION
16 # *****
17
18 TTF/8CL TC INTPRETX
19 DLOAD*
20 JDG2TTF,1
21 STODL* TABLTTF +6 # A(3) = 8 JDG TO TABLTTF
22 ADG2TTF,1 # 2
23 STODL TABLTTF +4 # A(2) = 6 ADG TO TABLTTF
24 VGU +4 # 2
25 DMP DAD*
26 3/4DP
27 VDG2TTF,1
28 STODL* TABLTTF +2 # A(1) = (6 VGU + 18 VDG)/8 TO TABLTTF
29 RDG +4,1 # 2 2
30 DSU DMP
31 RGU +4
32 3/8DP
33 STORE TABLTTF # A(0) = -24 (RGU - RDG)/64 TO TABLTTF
34 EXIT # 2 2
35
36 CA BIT8
37 TS TABLTTF +10 # FRACTIONAL PRECISION FOR TTF TO TABLE
38
39 EXTEND
40 DCA TTF/8
41 DXCH MPAC # LOADS TTF/8 (INITIAL GUESS) INTO MPAC
42 CAF TWO # DEGREE - ONE
43 TS L
44 CAF TABLTTF L
45 TC ROOTPSRS # YIELDS TTF/8 IN MPAC
46 INDEX WCHPHASE
47 TCF WHATALM
48
49 EXTEND # GOOD RETURN
50 DCA MPAC # FETCH TTF/8 KEEPING IT IN MPAC
51 DXCH TTF/8 # CORRECTED TTF/8
52
53
54
55
56
57
58
59
60
```

TC TDISPSET

# (CONTINUE TO QUADGUID)

# \*\*\*\*\*

# MAIN GUIDANCE EQUATION

# \*\*\*\*\*

#

# AS PUBLISHED --

#

# 
$$\overline{ACG} = \overline{ADG} + \frac{6(\overline{VDG} + \overline{VG})}{TTF} + \frac{12(\overline{RDG} - \overline{RG})}{(TTF)(TTF)}$$

#

# AS HERE PROGRAMMED --

#

# 
$$\overline{ACG} = \frac{3}{4} \left( \frac{1}{4}(\overline{RDG} - \overline{RG}) - \left( \frac{\overline{VDG} + \overline{VG}}{TTF/8} \right) \right) + \overline{ADG}$$

#

#

#

#

#

#

QUADGUID

CS TTF/8

AD LEADTIME

# LEADTIME IS A NEGATIVE NUMBER

AD POSMAX

# SAFEGUARD THE COMPUTATIONS THAT FOLLOW

TS L

# BY FORCING -TTF\*LEADTIME > OR = ZERO

CS L

AD L

ZL

EXTEND

DV TTF/8

TS BUF

# - RATIO OF LAG-DIMINISHED TTF TO TTF

EXTEND

SQUARE

TS BUF +1

AD BUF

XCH BUF +1

# RATIO SQUARED - RATIO

AD BUF +1

TS MPAC

# COEFFICIENT FOR VGU TERM

AD BUF +1

INDEX FIXLOC

TS 26D

# COEFFICIENT FOR RDG-RGU TERM

AD BUF +1

INDEX FIXLOC

TS 28D

# COEFFICIENT FOR VDG TERM

AD BUF

AD POSMAX

|    |          |       |           |                                                                                         |    |
|----|----------|-------|-----------|-----------------------------------------------------------------------------------------|----|
| 1  |          |       |           |                                                                                         | 1  |
| 2  |          | AD    | BUF +1    |                                                                                         | 2  |
| 3  |          | AD    | BUF +1    |                                                                                         | 3  |
| 4  |          | INDEX | FIXLOC    |                                                                                         | 4  |
| 5  |          | TS    | 30D       | # COEFFICIENT FOR ADG TERM                                                              | 5  |
| 6  |          |       |           |                                                                                         | 6  |
| 7  |          | CAF   | ZERO      |                                                                                         | 7  |
| 8  |          | TS    | MODE      |                                                                                         | 8  |
| 9  |          |       |           |                                                                                         | 9  |
| 10 |          | TC    | INTPRETX  |                                                                                         | 10 |
| 11 |          | VXSC  | PDDL      |                                                                                         | 11 |
| 12 |          |       | VGU       |                                                                                         | 12 |
| 13 |          |       | 28D       |                                                                                         | 13 |
| 14 |          | VXSC* | PDVL*     |                                                                                         | 14 |
| 15 |          |       | VDG,1     |                                                                                         | 15 |
| 16 |          |       | RDG,1     |                                                                                         | 16 |
| 17 |          | VSU   | V/SC      |                                                                                         | 17 |
| 18 |          |       | RGU       |                                                                                         | 18 |
| 19 |          |       | TTF/8     |                                                                                         | 19 |
| 20 |          | VSR2  | VXSC      |                                                                                         | 20 |
| 21 |          |       | 26D       |                                                                                         | 21 |
| 22 |          | VAD   | VAD       |                                                                                         | 22 |
| 23 |          | V/SC  | VXSC      |                                                                                         | 23 |
| 24 |          |       | TTF/8     |                                                                                         | 24 |
| 25 |          |       | 3/4DP     |                                                                                         | 25 |
| 26 |          | PDDL  | VXSC*     |                                                                                         | 26 |
| 27 |          |       | 30D       |                                                                                         | 27 |
| 28 |          |       | ADG,1     |                                                                                         | 28 |
| 29 |          | VAD   |           |                                                                                         | 29 |
| 30 | AFCCALC1 | VXM   | VSL1      | # VERGUID COMES HERE                                                                    | 30 |
| 31 |          |       | CG        |                                                                                         | 31 |
| 32 |          | PDVL  | V/SC      |                                                                                         | 32 |
| 33 |          |       | GDT/2     |                                                                                         | 33 |
| 34 |          |       | GSCALE    |                                                                                         | 34 |
| 35 |          | BVSU  | STADR     |                                                                                         | 35 |
| 36 |          | STORE | UNFC/2    | # UNFC/2 NEED NOT BE UNITIZED                                                           | 36 |
| 37 |          | ABVAL |           |                                                                                         | 37 |
| 38 | AFCCALC2 | STODL | /AFC/     | # MAGNITUDE OF AFC FOR THROTTLE                                                         | 38 |
| 39 |          |       | UNFC/2    | # VERTICAL COMPONENT                                                                    | 39 |
| 40 |          | DSQ   | PDDL      |                                                                                         | 40 |
| 41 |          |       | UNFC/2 +2 | # OUT-OF-PLANE                                                                          | 41 |
| 42 |          | DSQ   | PDDL      |                                                                                         | 42 |
| 43 |          |       | HIGHESTF  |                                                                                         | 43 |
| 44 |          | DDV   | DSQ       |                                                                                         | 44 |
| 45 |          |       | MASS      | #                                                                                       | 45 |
| 46 |          | DSU   | DSU       | # AMAXHORIZ = SQRT(ATOTAL - A <sup>2</sup> <sub>1</sub> - A <sup>2</sup> <sub>0</sub> ) | 46 |
| 47 |          | BPL   | DLOAD     | #                                                                                       | 47 |
| 48 |          |       | AFCCALC3  |                                                                                         | 48 |
| 49 | AFCCALC3 | SQRT  | ZEROVECS  |                                                                                         | 49 |
| 50 |          |       | DAD       |                                                                                         | 50 |
| 51 |          |       | UNFC/2 +4 |                                                                                         | 51 |
| 52 |          |       |           |                                                                                         | 52 |
| 53 |          |       |           |                                                                                         | 53 |
| 54 |          |       |           |                                                                                         | 54 |
| 55 |          |       |           |                                                                                         | 55 |
| 56 |          |       |           |                                                                                         | 56 |
| 57 |          |       |           |                                                                                         | 57 |
| 58 |          |       |           |                                                                                         | 58 |
| 59 |          |       |           |                                                                                         | 59 |
| 60 |          |       |           |                                                                                         | 60 |



|    |                                                      |        |            |    |
|----|------------------------------------------------------|--------|------------|----|
| 1  |                                                      |        |            | 1  |
| 2  |                                                      | BPL    | BDSU       | 2  |
| 3  |                                                      |        | AFCCLEND   | 3  |
| 4  | AFCCLEND                                             |        | UNFC/2 +4  | 4  |
| 5  |                                                      | STORE  | UNFC/2 +4  | 5  |
| 6  |                                                      | EXIT   |            | 6  |
| 7  |                                                      | TC     | FASTCHNG   | 7  |
| 8  |                                                      |        |            | 8  |
| 9  |                                                      | CA     | WCHPHASE   | 9  |
| 10 |                                                      | TS     | WCHPHOLD   | 10 |
| 11 |                                                      | INCR   | FLPASSO    | 11 |
| 12 |                                                      |        |            | 12 |
| 13 | BRSPOT4                                              | INDEX  | WCHPHASE   | 13 |
| 14 |                                                      | TCF    | AFTRGUID   | 14 |
| 15 |                                                      |        |            | 15 |
| 16 | # *****                                              |        |            | 16 |
| 17 | # ERECT GUIDANCE-STABLE MEMBER TRANSFORMATION MATRIX |        |            | 17 |
| 18 | # *****                                              |        |            | 18 |
| 19 |                                                      |        |            | 19 |
| 20 | CGCALC                                               | CAF    | EBANK5     | 20 |
| 21 |                                                      | TS     | EBANK      | 21 |
| 22 |                                                      | EBANK= | TCGIBRAK   | 22 |
| 23 |                                                      | EXTEND |            | 23 |
| 24 |                                                      | INDEX  | WCHPHASE   | 24 |
| 25 |                                                      | INDEX  | TARGETDEX  | 25 |
| 26 |                                                      | DCA    | TCGFBRAK   | 26 |
| 27 |                                                      | INCR   | BBANK      | 27 |
| 28 |                                                      | INCR   | BBANK      | 28 |
| 29 |                                                      | EBANK= | TTF/8      | 29 |
| 30 |                                                      | AD     | TTF/8      | 30 |
| 31 |                                                      | XCH    | L          | 31 |
| 32 |                                                      | AD     | TTF/8      | 32 |
| 33 |                                                      | CCS    | A          | 33 |
| 34 |                                                      | CCS    | L          | 34 |
| 35 |                                                      | TCF    | EXTLOGIC   | 35 |
| 36 |                                                      | TCF    | EXTLOGIC   | 36 |
| 37 |                                                      | NOOP   |            | 37 |
| 38 |                                                      |        |            | 38 |
| 39 |                                                      | TC     | INTPRETX   | 39 |
| 40 |                                                      | VLOAD  | UNIT       | 40 |
| 41 |                                                      |        | LAND       | 41 |
| 42 |                                                      | STODL  | CG         | 42 |
| 43 |                                                      |        | TTF/8      | 43 |
| 44 |                                                      | DMP*   | VXSC       | 44 |
| 45 |                                                      |        | GAINBRAK,1 | 45 |
| 46 |                                                      |        | ANGTERM    | 46 |
| 47 |                                                      | VAD    |            | 47 |
| 48 |                                                      |        | LAND       | 48 |
| 49 |                                                      | VSU    | RTB        | 49 |
| 50 |                                                      |        | R          | 50 |
| 51 |                                                      |        | NORMUNIT   | 51 |
| 52 |                                                      |        |            | 52 |
| 53 |                                                      |        |            | 53 |
| 54 |                                                      |        |            | 54 |
| 55 |                                                      |        |            | 55 |
| 56 |                                                      |        |            | 56 |
| 57 |                                                      |        |            | 57 |
| 58 |                                                      |        |            | 58 |
| 59 |                                                      |        |            | 59 |
| 60 |                                                      |        |            | 60 |

```
1
2 VXV RTB
3 LAND
4
5 STOVL NORMUNIT
6 CG +6 # SECOND ROW
7 CG
8
9 VXV VSL1
10 CG +6
11
12 STORE CG +14
13 EXIT
14
15 #
16 (CONTINUE TO EXTLOGIC)
17
18 #
19 # *****
20 # PREPARE TO EXIT
21 # *****
22 #
23 # DECIDE (1) HOW TO EXIT, AND (2) WHETHER TO SWITCH PHASES
24 #
25 EXTLOGIC INDEX WCHPHASE # WCHPHASE = 1 APPRQUAD
26 CA TENDBRAK # WCHPHASE = 0 BRAKQUAD
27 AD TTF/8
28
29 EXSPOT1 EXTEND
30 INDEX WCHPHASE
31 BZMF WHATEXIT
32
33 TC FASTCHNG
34
35 CA WCHPHOLD
36 AD ONE
37 TS WCHPHASE
38 CA ZERO
39 TS FLPASSO # RESET FLPASSO
40
41 INDEX WCHPHOLD
42 TCF WHATEXIT
43
44 # *****
45 # ROUTINES FOR EXITING FROM LANDING GUIDANCE
46 # *****
47 #
48 # 1. EXGSUB IS THE RETURN WHEN GUIDSUB IS CALLED BY THE IGNITION ALGORITHM.
49 # 2. EXBRAK IN THE EXIT USED DURING THE BRAKING PHASE. IN THIS CASE UNIT(R) IS THE WINDOW POINTING VECTOR.
50 # 3. EXNORM IS THE EXIT USED AT OTHER TIMES DURING THE BURN.
51 # (EXOVFLOW IS A SUBROUTINE OF EXBRAK AND EXNORM CALLED WHEN OVERFLOW OCCURRED ANYWHERE IN GUIDANCE.)
52
53 EXGSUB TC INTPRET # COMPUTE TRIM VELOCITY CORRECTION TERM.
```

|    |        |       |          |                                          |    |
|----|--------|-------|----------|------------------------------------------|----|
| 1  |        |       |          |                                          | 1  |
| 2  |        | VLOAD | RTB      |                                          | 2  |
| 3  |        |       | UNFC/2   |                                          | 3  |
| 4  |        |       | NORMUNIT |                                          | 4  |
| 5  |        | VXSC  | VXSC     |                                          | 5  |
| 6  |        |       | ZOOMTIME |                                          | 6  |
| 7  |        |       | TRIMACCL |                                          | 7  |
| 8  |        | STORE | UNFC/2   |                                          | 8  |
| 9  |        | EXIT  |          |                                          | 9  |
| 10 |        |       |          |                                          | 10 |
| 11 |        | CCS   | NGUIDSUB |                                          | 11 |
| 12 |        | TCF   | GUIDSUB  |                                          | 12 |
| 13 |        | CCS   | NIGNLOOP |                                          | 13 |
| 14 |        | TCF   | +3       |                                          | 14 |
| 15 |        | TC    | ALARM    |                                          | 15 |
| 16 |        | OCT   | 01412    |                                          | 16 |
| 17 |        |       |          |                                          | 17 |
| 18 | +3     | TC    | POSTJUMP |                                          | 18 |
| 19 |        | CADR  | DDUMCALC |                                          | 19 |
| 20 |        |       |          |                                          | 20 |
| 21 | EXBRAK | TC    | INTPRET  |                                          | 21 |
| 22 |        | VLOAD |          |                                          | 22 |
| 23 |        |       | UNIT/R/  |                                          | 23 |
| 24 |        | STORE | UNWC/2   |                                          | 24 |
| 25 |        | EXIT  |          |                                          | 25 |
| 26 |        | TCF   | STEER?   |                                          | 26 |
| 27 |        |       |          |                                          | 27 |
| 28 | EXNORM | TC    | INTPRET  |                                          | 28 |
| 29 |        | VLOAD | VSU      |                                          | 29 |
| 30 |        |       | LAND     |                                          | 30 |
| 31 |        |       | R        |                                          | 31 |
| 32 |        | RTB   |          |                                          | 32 |
| 33 |        |       | NORMUNIT |                                          | 33 |
| 34 |        | STORE | UNWC/2   | # UNIT(LAND - R) IS TENTATIVE CHOICE     | 34 |
| 35 |        | VXV   | DOT      |                                          | 35 |
| 36 |        |       | XNBPIP   |                                          | 36 |
| 37 |        |       | CG +6    |                                          | 37 |
| 38 |        | EXIT  |          | # WITH PROJ IN MPAC 1/8 REAL SIZE        | 38 |
| 39 |        |       |          |                                          | 39 |
| 40 |        | CS    | MPAC     | # GET COEFFICIENT FOR CG +14             | 40 |
| 41 |        | AD    | PROJMAX  |                                          | 41 |
| 42 |        | AD    | POSMAX   |                                          | 42 |
| 43 |        | TS    | BUF      |                                          | 43 |
| 44 |        | CS    | BUF      |                                          | 44 |
| 45 |        | ADS   | BUF      | # RESULT IS 0 IF PROJMAX - PROJ NEGATIVE | 45 |
| 46 |        |       |          |                                          | 46 |
| 47 |        | CS    | PROJMIN  | # GET COEFFICIENT FOR UNIT(LAND - R)     | 47 |
| 48 |        | AD    | MPAC     |                                          | 48 |
| 49 |        | AD    | POSMAX   |                                          | 49 |
| 50 |        | TS    | BUF +1   |                                          | 50 |
| 51 |        | CS    | BUF +1   |                                          | 51 |
| 52 |        |       |          |                                          | 52 |
| 53 |        |       |          |                                          | 53 |
| 54 |        |       |          |                                          | 54 |
| 55 |        |       |          |                                          | 55 |
| 56 |        |       |          |                                          | 56 |
| 57 |        |       |          |                                          | 57 |
| 58 |        |       |          |                                          | 58 |
| 59 |        |       |          |                                          | 59 |
| 60 |        |       |          |                                          | 60 |

|    |           |        |          |                                          |    |
|----|-----------|--------|----------|------------------------------------------|----|
| 1  |           |        |          |                                          | 1  |
| 2  |           | ADS    | BUF +1   | # RESULT IS 0 IF PROJ - PROJMIN NEGATIVE | 2  |
| 3  |           |        |          |                                          | 3  |
| 4  |           | CAF    | FOUR     |                                          | 4  |
| 5  | UNWCLOOP  | MASK   | SIX      |                                          | 5  |
| 6  |           | TS     | Q        |                                          | 6  |
| 7  |           | CA     | EBANK5   |                                          | 7  |
| 8  |           | TS     | EBANK    |                                          | 8  |
| 9  |           | EBANK= | CG       |                                          | 9  |
| 10 |           | CA     | BUF      |                                          | 10 |
| 11 |           | EXTEND |          |                                          | 11 |
| 12 |           | INDEX  | Q        |                                          | 12 |
| 13 |           | MP     | CG +14   |                                          | 13 |
| 14 |           | INCR   | BBANK    |                                          | 14 |
| 15 |           | EBANK= | UNWC/2   |                                          | 15 |
| 16 |           | INDEX  | Q        |                                          | 16 |
| 17 |           | DXCH   | UNWC/2   |                                          | 17 |
| 18 |           | EXTEND |          |                                          | 18 |
| 19 |           | MP     | BUF +1   |                                          | 19 |
| 20 |           | INDEX  | Q        |                                          | 20 |
| 21 |           | DAS    | UNWC/2   |                                          | 21 |
| 22 |           | CCS    | Q        |                                          | 22 |
| 23 |           | TCF    | UNWCLOOP |                                          | 23 |
| 24 |           |        |          |                                          | 24 |
| 25 |           | INCR   | BBANK    |                                          | 25 |
| 26 |           | EBANK= | PIF      |                                          | 26 |
| 27 |           |        |          |                                          | 27 |
| 28 | STEER?    | CA     | FLAGWRD2 | # IF STEERSW DOWN NO OUTPUTS             | 28 |
| 29 |           | MASK   | STEERBIT |                                          | 29 |
| 30 |           | EXTEND |          |                                          | 30 |
| 31 |           | BZF    | RATESTOP |                                          | 31 |
| 32 |           |        |          |                                          | 32 |
| 33 | EXVERT    | CA     | OVFIND   | # IF OVERFLOW ANYWHERE IN GUIDANCE       | 33 |
| 34 |           | EXTEND |          | # DON'T CALL THROTTLE OR FINDCDUW        | 34 |
| 35 |           | BZF    | +13      |                                          | 35 |
| 36 |           |        |          |                                          | 36 |
| 37 | EXOVLFLOW | TC     | ALARM    | # SOUND THE ALARM NON-ABORTIVELY         | 37 |
| 38 |           | OCT    | 01410    |                                          | 38 |
| 39 |           |        |          |                                          | 39 |
| 40 | RATESTOP  | CAF    | BIT13    | # ARE WE IN ATTITUDE-HOLD?               | 40 |
| 41 |           | EXTEND |          |                                          | 41 |
| 42 |           | RAND   | CHAN31   |                                          | 42 |
| 43 |           | EXTEND |          |                                          | 43 |
| 44 |           | BZF    | DISPEXIT | # YES                                    | 44 |
| 45 |           |        |          |                                          | 45 |
| 46 |           | TC     | BANKCALL | # NO: DO A STOPRATE                      | 46 |
| 47 |           | CADR   | STOPRATE |                                          | 47 |
| 48 |           |        |          |                                          | 48 |
| 49 |           | TCF    | DISPEXIT |                                          | 49 |
| 50 |           |        |          |                                          | 50 |
| 51 | GDUMP1    | TC     | THROTTLE |                                          | 51 |
| 52 |           |        |          |                                          | 52 |
| 53 |           |        |          |                                          | 53 |
| 54 |           |        |          |                                          | 54 |
| 55 |           |        |          |                                          | 55 |
| 56 |           |        |          |                                          | 56 |
| 57 |           |        |          |                                          | 57 |
| 58 |           |        |          |                                          | 58 |
| 59 |           |        |          |                                          | 59 |
| 60 |           |        |          |                                          | 60 |

```
1 TC INTPRET
2 CALL
3
4 EXIT FINDCDUW -2
5
6 #
7 (CONTINUE TO DISPEXIT)
8
9 # *****
10 # GUIDANCE LOOP DISPLAYS
11 # *****
12
13 DISPEXIT EXTEND # KILL GROUP 3: DISPLAYS WILL BE
14 DCA NEG0 # RESTORED BY NEXT GUIDANCE CYCLE.
15 DXCH -PHASE3
16
17 +3 CS FLAGWRD8 # IF FLUNDISP IS SET, NO DISPLAY THIS PASS
18 MASK FLUNDBIT
19
20 EXTEND
21 BZF ENDLLJOB # TO PICK UP THE TAG
22
23 INDEX WCHPHOLD
24 TCF WHATDISP
25
26 -2 TC PHASCHNG # KILL GROUP 5
27 OCT 00035
28
29 P63DISPS CAF V06N63
30 DISPCOMN TC BANKCALL
31 CADR REGODSPR
32
33 ENDLLJOB TCF ENDOFJOB
34
35 P64DISPS CA TREDES # HAS TREDES REACHED ZERO?
36 EXTEND
37 BZF RED-OVER # YES: CLEAR REDESIGNATION FLAG
38
39 CS FLAGWRD6 # NO: IS REDFLAG SET?
40 MASK REDFLBIT
41
42 EXTEND
43 BZF REDES-OK # YES: DO STATIC DISPLAY
44
45 CAF V06N64 # OTHERWISE USE FLASHING DISPLAY
46 TC BANKCALL
47 CADR REFLASHR
48 TCF GOTOPOOH # TERMINATE
49 TCF P64CEED # PROCEED PERMIT REDESIGNATIONS
50 TCF P64DISPS # RECYCLE
51
52
53
54
55
56
57
58
59
60
```

|    |          |                                            |                                    |                              |    |
|----|----------|--------------------------------------------|------------------------------------|------------------------------|----|
| 1  |          |                                            |                                    |                              | 1  |
| 2  |          | TCF                                        | ENDLLJOB                           |                              | 2  |
| 3  |          |                                            |                                    |                              | 3  |
| 4  | P64CEED  | CAF                                        | ZERO                               |                              | 4  |
| 5  |          | TS                                         | ELINCR1                            |                              | 5  |
| 6  |          | TS                                         | AZINCR1                            |                              | 6  |
| 7  |          |                                            |                                    |                              | 7  |
| 8  |          | TC                                         | UPFLAG                             | # ENABLE REDESIGNATION LOGIC | 8  |
| 9  |          | ADRES                                      | REDFLAG                            |                              | 9  |
| 10 |          |                                            |                                    |                              | 10 |
| 11 |          | TCF                                        | ENDOFJOB                           |                              | 11 |
| 12 |          |                                            |                                    |                              | 12 |
| 13 | RED-OVER | TC                                         | DOWNFLAG                           |                              | 13 |
| 14 |          | ADRES                                      | REDFLAG                            |                              | 14 |
| 15 | REDES-OK | CAF                                        | V06N64                             |                              | 15 |
| 16 |          | TCF                                        | DISPCOMN                           |                              | 16 |
| 17 |          |                                            |                                    |                              | 17 |
| 18 |          |                                            |                                    |                              | 18 |
| 19 | VERTDISP | CAF                                        | V06N60                             |                              | 19 |
| 20 |          | TCF                                        | DISPCOMN                           |                              | 20 |
| 21 |          |                                            |                                    |                              | 21 |
| 22 |          |                                            |                                    |                              | 22 |
| 23 | #        | *****                                      |                                    |                              | 23 |
| 24 | #        | GUIDANCE FOR P65                           |                                    |                              | 24 |
| 25 | #        | *****                                      |                                    |                              | 25 |
| 26 |          |                                            |                                    |                              | 26 |
| 27 | VERTGUID | CCS                                        | WCHVERT                            |                              | 27 |
| 28 |          | TCF                                        | P67VERT                            | # POSITIVE NON-ZERO ---> P67 | 28 |
| 29 |          | TCF                                        | P66VERT                            | # +0                         | 29 |
| 30 | #        |                                            |                                    |                              | 30 |
| 31 | #        | THE P65 GUIDANCE EQUATION IS AS FOLLOWS -- |                                    |                              | 31 |
| 32 | #        |                                            |                                    |                              | 32 |
| 33 | #        |                                            | $\overline{V2FG} - \overline{VGU}$ |                              | 33 |
| 34 | #        | ACG =                                      | -----                              |                              | 34 |
| 35 | #        |                                            | TAUVERT                            |                              | 35 |
| 36 |          |                                            |                                    |                              | 36 |
| 37 | P65VERT  | TC                                         | INTPRET                            |                              | 37 |
| 38 |          | VLOAD                                      | VSU                                |                              | 38 |
| 39 |          |                                            | V2FG                               |                              | 39 |
| 40 |          |                                            | VGU                                |                              | 40 |
| 41 |          | V/SC                                       | GOTO                               |                              | 41 |
| 42 |          |                                            | TAUVERT                            |                              | 42 |
| 43 |          |                                            | AFCCALC1                           |                              | 43 |
| 44 |          |                                            |                                    |                              | 44 |
| 45 |          |                                            |                                    |                              | 45 |
| 46 |          |                                            |                                    |                              | 46 |
| 47 |          |                                            |                                    |                              | 47 |
| 48 |          |                                            |                                    |                              | 48 |
| 49 |          |                                            |                                    |                              | 49 |
| 50 |          |                                            |                                    |                              | 50 |
| 51 |          |                                            |                                    |                              | 51 |
| 52 |          |                                            |                                    |                              | 52 |
| 53 |          |                                            |                                    |                              | 53 |
| 54 |          |                                            |                                    |                              | 54 |
| 55 |          |                                            |                                    |                              | 55 |
| 56 |          |                                            |                                    |                              | 56 |
| 57 |          |                                            |                                    |                              | 57 |
| 58 |          |                                            |                                    |                              | 58 |
| 59 |          |                                            |                                    |                              | 59 |
| 60 |          |                                            |                                    |                              | 60 |

# \*\*\*\*\*  
# GUIDANCE FOR P66

# \*\*\*\*\*

P66VERT TC POSTJUMP  
CADR P66VERTA

P67VERT TC PHASCHNG # TERMINATE GROUP 3.  
OCT 00003

TC INTPRET  
VLOAD GOTO  
V  
VHORCOMP

SETLOC P66LOC  
BANK  
COUNT\* \$\$/F2DPS

RODTASK CAF PRI022  
TC FINDVAC  
EBANK= DVCNTR  
2CADR RODCOMP

TCF TASKOVER

P66VERTA TC PHASCHNG # TERMINATE GROUP 3.  
OCT 00003

CAF 1SEC  
TC TWIDDLE  
ADRES RODTASK

RODCOMP INHINT  
CAF ZERO  
XCH RODCOUNT  
EXTEND  
MP RODSCAL1

DAS VDGVERT # UPDATE DESIRED ALTITUDE RATE.

EXTEND # SET OLDPIPAX,Y,Z = PIPAX,Y,Z

DCA PIPAX  
DXCH OLDPIPAX  
DXCH RUPTREG1 # SET RUPTREG1,2,3 = OLDPIPAX,Y,Z

CA PIPAZ  
XCH OLDPIPAZ  
XCH RUPTREG3

EXTEND # SNAPSHOT TIME OF PIPA READING.  
DCA TIME2

|    |   |
|----|---|
| 76 | 1 |
| 77 |   |





|                   |                        |                                         |        |                                          |
|-------------------|------------------------|-----------------------------------------|--------|------------------------------------------|
|                   | DAD                    | PDDL<br>OD                              | #      | (4)                                      |
|                   | DDV                    | 20D<br>DSU<br>22D                       | #      | (2)                                      |
|                   | DMP                    | DAD<br>LAG/TAU<br>/AFC/                 |        |                                          |
|                   | PDDL                   | DDV<br>MAXFORCE<br>MASS                 | #      | (4)                                      |
|                   | PDDL                   | DDV<br>MINFORCE<br>MASS                 | #      | (6)                                      |
|                   | PUSH                   | BDSU<br>2D                              | #      | (8)                                      |
|                   | BMN                    | DLOAD                                   | #      | (6)                                      |
|                   | DLOAD<br>BDSU          | AFCSPOT<br>PUSH<br>BPL<br>2D<br>AFCSPOT | #      | (6)                                      |
|                   | DLOAD                  |                                         | #      | (4)                                      |
| AFCSPOT           | DLOAD<br>SETPD         |                                         | #<br># | (2), (4), OR (6)<br>(2)                  |
|                   |                        | 2D                                      |        |                                          |
| ITRPNT2           | STODL<br>EXIT<br>DXCH  | /AFC/<br>MPAC                           | #      | (0)<br># MPAC = MEASURED ACCELERATION.   |
|                   | TC<br>CADR<br>TC       | BANKCALL<br>THROTTLE +3<br>INTPRET      |        |                                          |
|                   | VLOAD                  |                                         | #      | PICK UP UPDATED VELOCITY VECTOR.         |
| VHORCOMP          | VSL2                   | 24D<br>VAD                              |        |                                          |
|                   | VSR2                   | DELVS<br>PDVL<br>R                      |        |                                          |
|                   | UNIT                   | VXSC<br>HDOTDISP<br>BVSU                |        |                                          |
|                   | ABVAL<br>STORE<br>EXIT | VHORIZ                                  |        |                                          |
|                   | TC<br>CADR             | BANKCALL<br>DISPEXIT +3                 | #      | PUT UP V06N60 DISPLAY BUT AVOID PHASCHNG |
| BIT1H<br>SHFTFACT | OCT<br>2DEC            | 00001<br>1 B-17                         |        |                                          |

BIASFACT 2DEC 655.36 B-28

# \*\*\*\*\*  
# REDESIGNATOR TRAP  
# \*\*\*\*\*

BANK 11  
SETLOC F2DPS\*11  
BANK

COUNT\* \$\$/F2DPS

PITFALL XCH BANKRUPT  
EXTEND  
QXCH QRUPT

TC CHECKMM # IF NOT IN P64, NO REASON TO CONTINUE  
DEC 64  
TCF RESUME

EXTEND  
READ CHAN31  
COM

MASK ALL4BITS  
TS ELVIRA  
CAF TWO

TS ZERLINA  
CAF FIVE  
TC TWIDDLE

ADRES REDESMON  
TCF RESUME

# REDESIGNATOR MONITOR (INITIATED BY PITFALL)

PREMON1 TS ZERLINA  
PREMON2 CAF SEVEN  
TC VARDELAY

REDESMON EXTEND  
READ 31  
COM

MASK ALL4BITS  
XCH ELVIRA  
TS L

CCS ELVIRA # DO ANY BITS APPEAR THIS PASS?  
TCF PREMON2 # Y: CONTINUE MONITOR

CCS L # N: ANY LAST PASS?  
TCF COUNT 'EM # Y: COUNT 'EM, RESET RUPT, TERMINATE



```
1 ALL4BITS OCT 00063
2 AZEACH DEC .03491 # 2 DEGREES
3 ELEACH DEC .00873 # 1/2 DEGREE
```

```
4 # *****
```

```
5 # R.O.D. TRAP
```

```
6 # *****
```

```
7
8 BANK 20
9 SETLOC RODTRAP
10 BANK
```

```
11 COUNT* $$/F2DPS # *****
```

```
12
13 DESCBITS MASK BIT7 # COME HERE FROM MARKRUPT CODING WITH BIT
14 CCS A # 7 OR 6 OF CHANNEL 16 IN A; BIT 7 MEANS
15 CS TWO # - RATE INCREMENT, BIT 6 + INCREMENT.
16 AD ONE
```

```
17 ADS RODCOUNT
18 TCF RESUME # TRAP IS RESET WHEN SWITCH IS RELEASED
```

```
19
20 BANK 31
21 SETLOC F2DPS*31
22 BANK
```

```
23 COUNT* $$/F2DPS
```

```
24 # *****
```

```
25 # DOUBLE PRECISION ROOT FINDER SUBROUTINE (BY ALLAN KLUMPP)
```

```
26 # *****
```

```
27 #
28 #
29 # N N-1
30 # ROOTPSRS FINDS ONE ROOT OF THE POWER SERIES A X + A X + ... + A X + A
```

```
31 # N N-1 1 0
32 # USING NEWTON'S METHOD STARTING WITH AN INITIAL GUESS FOR THE ROOT. THE ENTERING DATA MUST BE AS FOLLOWS:
```

```
33 # A SP LOC-3 ADRES FOR REFERENCING PWR COF TABL
```

```
34 # L SP N-1 N IS THE DEGREE OF THE POWER SERIES
```

```
35 # MPAC DP X INITIAL GUESS FOR ROOT
```

```
36 #
37 # LOC-2N DP A(0)
```

```
38 # #
39 # LOC DP ...
```

```
40 # LOC DP A(N)
```

```
41 # LOC+2 SP PRECROOT PREC RQD OF ROOT (AS FRACT OF 1ST GUESS)
```

```
42 #
```

# THE DP RESULT IS LEFT IN MPAC UPON EXIT, AND A SP COUNT OF THE ITERATIONS TO CONVERGENCE IS LEFT IN MPAC+2.  
# RETURN IS NORMALLY TO LOC(TC ROOTPSRS)+3. IF ROOTPSRS FAILS TO CONVERGE TO IN 8 PASSES, RETURN IS TO LOC+1 AND  
# OUTPUTS ARE NOT TO BE TRUSTED.  
#  
# PRECAUTION: ROOTPSRS MAKES NO CHECKS FOR OVERFLOW OR FOR IMPROPER USAGE. IMPROPER USAGE COULD  
# PRECLUDE CONVERGENCE OR REQUIRE EXCESSIVE ITERATIONS. AS A SPECIFIC EXAMPLE, ROOTPSRS FORMS A DERIVATIVE  
# COEFFICIENT TABLE BY MULTIPLYING EACH A(I) BY I, WHERE I RANGES FROM 1 TO N. IF AN ELEMENT OF THE DERIVATIVE  
# COEFFICIENT TABLE = 1 OR >1 IN MAGNITUDE, ONLY THE EXCESS IS RETAINED. ROOTPSRS MAY CONVERGE ON THE CORRECT  
# ROOT NONETHELESS, BUT IT MAY TAKE AN EXCESSIVE NUMBER OF ITERATIONS. THEREFORE THE USER SHOULD RECOGNIZE:  
# 1. USER'S RESPONSIBILITY TO ASSUR THAT  $I \times A(I) < 1$  IN MAGNITUDE FOR ALL I.  
# 2. USER'S RESPONSIBILITY TO ASSURE OVERFLOW WILL NOT OCCUR IN EVALUATING EITHER THE RESIDUAL OR THE DERIVATIVE  
# POWER SERIES. THIS OVERFLOW WOULD BE PRODUCED BY SUBROUTINE POWRSRS, CALLED BY ROOTPSRS, AND MIGHT NOT  
# PRECLUDE EVENTUAL CONVERGENCE.  
# 3. AT PRESENT, ERASABLE LOCATIONS ARE RESERVED ONLY FOR N UP TO 5. AN N IN EXCESS OF 5 WILL PRODUCE CHAOS.  
# ALL ERASABLES USED BY ROOTPSRS ARE UNSWITCHED LOCATED IN THE REGION FROM MPAC-33 OCT TO MPAC+7.  
# 4. THE ITERATION COUNT RETURNED IN MPAC+2 MAY BE USED TO DETECT ABNORMAL PERFORMANCE.

|          |        |          |                                             |
|----------|--------|----------|---------------------------------------------|
| ROOTPSRS | EXTEND |          | # STORE ENTERING DATA, INITIALIZE ERASABLES |
|          | QXCH   | RETROOT  | # RETURN ADRES                              |
|          | TS     | PWRPTR   | # PWR TABLE POINTER                         |
|          | DXCH   | MPAC +3  | # PWR TABLE ADRES, N-1                      |
|          | CA     | DERTABLL |                                             |
|          | TS     | DERPTR   | # DER TABL POINTER                          |
|          | TS     | MPAC +5  | # DER TABL ADRES                            |
|          | CCS    | MPAC +4  | # NO POWER SERIES DEGREE 1 OR LESS          |
|          | TS     | MPAC +6  | # N-2                                       |
|          | CA     | ZERO     | # MODE USED AS ITERATION COUNTER. MODE      |
|          | TS     | MODE     | # MUST BE POS SO ABS WON'T COMP MPAC+3 ETC. |

# COMPUTE CRITERION TO STOP ITERATING

EXTEND

|       |          |                                           |
|-------|----------|-------------------------------------------|
| DCA   | MPAC     | # FETCH ROOT GUESS, KEEPING IT IN MPAC    |
| DXCH  | ROOTPS   | # AND IN ROOTPS                           |
| INDEX | MPAC +3  | # PWR TABLE ADRES                         |
| CA    | 5        | # PRECROOT TO A                           |
| TC    | SHORTMP  | # YIELDS DP PRODUCT IN MPAC               |
| TC    | USPRCADR |                                           |
| CADR  | ABS      | # YIELDS ABVAL OF CRITERION ON DX IN MPAC |
| DXCH  | MPAC     |                                           |
| DXCH  | DXCRIT   | # CRITERION                               |

# SET UP DER COF TABL

|          |          |                                             |                |
|----------|----------|---------------------------------------------|----------------|
| EXTEND   |          |                                             |                |
| INDEX    | PWRPTR   |                                             |                |
| DCA      | 3        |                                             |                |
| DXCH     | MPAC     | # A(N) TO MPAC                              |                |
| CA       | MPAC +4  | # N-1 TO A                                  |                |
| DERCLOOP | TS       | PWRCNT                                      | # LOOP COUNTER |
| AD       | ONE      |                                             |                |
| TC       | DMPNSUB  | # YIELDS DERCOF = I X A(I) IN MPAC          |                |
| EXTEND   |          |                                             |                |
| INDEX    | PWRPTR   |                                             |                |
| DCA      | 1        |                                             |                |
| DXCH     | MPAC     | # (I-1) TO MPAC, FETCHING DERCOF            |                |
| INDEX    | DERPTR   |                                             |                |
| DXCH     | 3        | # DERCOF TO DER TABLE                       |                |
| CS       | TWO      |                                             |                |
| ADS      | PWRPTR   | # DECREMENT PWR POINTER                     |                |
| CS       | TWO      |                                             |                |
| ADS      | DERPTR   | # DECREMENT DER POINTER                     |                |
| CCS      | PWRCNT   |                                             |                |
| TCF      | DERCLOOP |                                             |                |
| ROOTLOOP | EXTEND   | # CONVERGE ON ROOT                          |                |
| DCA      | ROOTPS   | # FETCH CURRENT ROOT                        |                |
| DXCH     | MPAC     | # LEAVE IN MPAC                             |                |
| EXTEND   |          |                                             |                |
| DCA      | MPAC +5  | # LOAD A, L WITH DER TABL ADRES, N-2        |                |
| TC       | POWRSERS | # YIELDS DERIVATIVE IN MPAC                 |                |
| EXTEND   |          |                                             |                |
| DCA      | ROOTPS   |                                             |                |
| DXCH     | MPAC     | # CURRENT ROOT TO MPAC, FETCHING DERIVATIVE |                |
| DXCH     | BUF      | # LEAVE DERIVATIVE IN BUF AS DIVISOR        |                |
| EXTEND   |          |                                             |                |
| DCA      | MPAC +3  | # LOAD A, L WITH PWR TABL ADRES, N-1        |                |
| TC       | POWRSERS | # YIELDS RESIDUAL IN MPAC                   |                |
| TC       | USPRCADR |                                             |                |
| CADR     | DDV/BDDV | # YIELDS -DX IN MPAC                        |                |
| EXTEND   |          |                                             |                |
| DCS      | MPAC     | # FETCH DX, LEAVING -DX IN MPAC             |                |
| DAS      | ROOTPS   | # CORRECTED ROOT NOW IN ROOTPS              |                |
| TC       | USPRCADR |                                             |                |
| CADR     | ABS      | # YIELDS ABS(DX) IN MPAC                    |                |
| EXTEND   |          |                                             |                |

```
1
2 DCS DXCRIT
3 DAS MPAC # ABS(DX)-ABS(DXCRIT) IN MPAC
4
5 CA MODE
6 MASK BIT4 # KLUMPP SAYS GIVE UP AFTER EIGHT PASSES
7
8 BADROOT CCS A
9 TC RETROOT
10
11 INCR MODE # INCREMENT ITERATION COUNTER
12 CCS MPAC # TEST HI ORDER DX
13 TCF ROOTLOOP
14
15 TESTLODX TCF TESTLODX
16 TCF ROOTSTOR
17 CCS MPAC +1 # TEST LO ORDER DX
18
19 ROOTSTOR TCF ROOTLOOP
20 DXCH ROOTSTOR
21 DXCH ROOTSTOR
22 CA ROOTSTOR
23 TS ROOTPS
24 INDEX MPAC
25 TCF MPAC +2 # STORE SP ITERATION COUNT IN MPAC+2
26
27 DERTABLL ADRES DERCOFN -3
28
29 # *****
30 # TRASHY LITTLE SUBROUTINES
31 # *****
32
33 INTPRETX INDEX WCHPHASE # SET X1 ON THE WAY TO THE INTERPRETER
34 CS TARGETDEX
35
36 INDEX FIXLOC
37 TS X1
38 TCF INTPRET
39
40 TDISPSET CA TTF/8
41 EXTEND
42
43 MP TSCALINV
44 DXCH TTFDISP
45
46 CA EBANK5 # TREDES BECOMES ZERO TWO PASSES
47 TS EBANK # BEFORE TCGFAPPR IS REACHED
48 EBANK= TCGFAPPR
49
50 CA TCGFAPPR
51 INCR BBANK
52 INCR BBANK
53 EBANK= TTF/8
54
55
56
57
58
59
60
```



```

1
2 AD TTF/8
3 EXTEND
4 MP TREDESCL
5 AD -DEC103
6 AD NEGMAX
7 TS L
8 CS L
9 AD L
10 AD +DEC99
11 AD POSMAX
12 TS TREDES
13 CS TREDES
14 ADS TREDES
15 TC Q
16
17 1406P00 TC POOD00
18 OCT 01406
19 1406ALM TC ALARM
20 OCT 01406
21 TCF RATESTOP
22
23 # *****
24 # SPECIALIZED "PHASCHNG" SUBROUTINE
25 # *****
26
27 EBANK= PHSNAME2
28 FASTCHNG CA EBANK3 # SPECIALIZED 'PHASCHNG' ROUTINE
29 XCH EBANK
30 DXCH L
31 TS PHSNAME3
32 LXCH EBANK
33 EBANK= E2DPS
34 TC A
35
36 # *****
37 # PARAMETER TABLE INDIRECT ADDRESSES
38 # *****
39
40 RDG = RBRFG
41 VDG = VBRFG
42 ADG = ABRFG
43 VDG2TTF = VBRFG*
44 ADG2TTF = ABRFG*
45 JDG2TTF = JBRFG*
46
47 # *****
48 # LUNAR LANDING CONSTANTS
49 # *****
50
51
52
53
54
55
56
57
58
59
60
```



|    |          |        |                 |                                     |  |    |
|----|----------|--------|-----------------|-------------------------------------|--|----|
| 1  |          |        |                 |                                     |  | 1  |
| 2  | TABLTTFL | ADRES  | TABLTTF +3      | # ADDRESS FOR REFERENCING TTF TABLE |  | 2  |
| 3  | TTFSCALE | =      | BIT12           |                                     |  | 3  |
| 4  | TSCALINV | =      | BIT4            |                                     |  | 4  |
| 5  | -DEC103  | DEC    | -103            |                                     |  | 5  |
| 6  | +DEC99   | DEC    | +99             |                                     |  | 6  |
| 7  | TREDESCL | DEC    | -.08            |                                     |  | 7  |
| 8  | 180DEGS  | DEC    | +180            |                                     |  | 8  |
| 9  | 1/2DEG   | DEC    | +.00278         |                                     |  | 9  |
| 10 | PROJMAX  | DEC    | .42262 B-3      | # SIN(25')/8 TO COMPARE WITH PROJ   |  | 10 |
| 11 | PROJMIN  | DEC    | .25882 B-3      | # SIN(15')/8 TO COMPARE WITH PROJ   |  | 11 |
| 12 | V06N63   | VN     | 0663            | # P63                               |  | 12 |
| 13 | V06N64   | VN     | 0664            | # P64                               |  | 13 |
| 14 | V06N60   | VN     | 0660            | # P65, P66, P67                     |  | 14 |
| 15 |          |        |                 |                                     |  | 15 |
| 16 |          | BANK   | 22              |                                     |  | 16 |
| 17 |          | SETLOC | LANDCNST        |                                     |  | 17 |
| 18 |          | BANK   |                 |                                     |  | 18 |
| 19 |          | COUNT* | \$\$/F2DPS      |                                     |  | 19 |
| 20 |          |        |                 |                                     |  | 20 |
| 21 | HIGHESTF | 2DEC   | 4.34546769 B-12 |                                     |  | 21 |
| 22 | GSCALE   | 2DEC   | 100 B-11        |                                     |  | 22 |
| 23 | 3/8DP    | 2DEC   | .375            |                                     |  | 23 |
| 24 | 3/4DP    | 2DEC   | .750            |                                     |  | 24 |
| 25 | DEPRCRIT | 2DEC   | -.02 B-1        |                                     |  | 25 |
| 26 |          |        |                 |                                     |  | 26 |
| 27 |          |        |                 |                                     |  | 27 |
| 28 |          |        |                 |                                     |  | 28 |
| 29 |          |        |                 |                                     |  | 29 |
| 30 |          |        |                 |                                     |  | 30 |
| 31 |          |        |                 |                                     |  | 31 |
| 32 |          |        |                 |                                     |  | 32 |
| 33 |          |        |                 |                                     |  | 33 |
| 34 |          |        |                 |                                     |  | 34 |
| 35 |          |        |                 |                                     |  | 35 |
| 36 |          |        |                 |                                     |  | 36 |
| 37 |          |        |                 |                                     |  | 37 |
| 38 |          |        |                 |                                     |  | 38 |
| 39 |          |        |                 |                                     |  | 39 |
| 40 |          |        |                 |                                     |  | 40 |
| 41 |          |        |                 |                                     |  | 41 |
| 42 |          |        |                 |                                     |  | 42 |
| 43 |          |        |                 |                                     |  | 43 |
| 44 |          |        |                 |                                     |  | 44 |
| 45 |          |        |                 |                                     |  | 45 |
| 46 |          |        |                 |                                     |  | 46 |
| 47 |          |        |                 |                                     |  | 47 |
| 48 |          |        |                 |                                     |  | 48 |
| 49 |          |        |                 |                                     |  | 49 |
| 50 |          |        |                 |                                     |  | 50 |
| 51 |          |        |                 |                                     |  | 51 |
| 52 |          |        |                 |                                     |  | 52 |
| 53 |          |        |                 |                                     |  | 53 |
| 54 |          |        |                 |                                     |  | 54 |
| 55 |          |        |                 |                                     |  | 55 |
| 56 |          |        |                 |                                     |  | 56 |
| 57 |          |        |                 |                                     |  | 57 |
| 58 |          |        |                 |                                     |  | 58 |
| 59 |          |        |                 |                                     |  | 59 |
| 60 |          |        |                 |                                     |  | 60 |



|    |         |    |
|----|---------|----|
| 1  |         | 1  |
| 2  | # ***** | 2  |
| 3  | # ***** | 3  |
| 4  |         | 4  |
| 5  |         | 5  |
| 6  |         | 6  |
| 7  |         | 7  |
| 8  |         | 8  |
| 9  |         | 9  |
| 10 |         | 10 |
| 11 |         | 11 |
| 12 |         | 12 |
| 13 |         | 13 |
| 14 |         | 14 |
| 15 |         | 15 |
| 16 |         | 16 |
| 17 |         | 17 |
| 18 |         | 18 |
| 19 |         | 19 |
| 20 |         | 20 |
| 21 |         | 21 |
| 22 |         | 22 |
| 23 |         | 23 |
| 24 |         | 24 |
| 25 |         | 25 |
| 26 |         | 26 |
| 27 |         | 27 |
| 28 |         | 28 |
| 29 |         | 29 |
| 30 |         | 30 |
| 31 |         | 31 |
| 32 |         | 32 |
| 33 |         | 33 |
| 34 |         | 34 |
| 35 |         | 35 |
| 36 |         | 36 |
| 37 |         | 37 |
| 38 |         | 38 |
| 39 |         | 39 |
| 40 |         | 40 |
| 41 |         | 41 |
| 42 |         | 42 |
| 43 |         | 43 |
| 44 |         | 44 |
| 45 |         | 45 |
| 46 |         | 46 |
| 47 |         | 47 |
| 48 |         | 48 |
| 49 |         | 49 |
| 50 |         | 50 |
| 51 |         | 51 |
| 52 |         | 52 |
| 53 |         | 53 |
| 54 |         | 54 |
| 55 |         | 55 |
| 56 |         | 56 |
| 57 |         | 57 |
| 58 |         | 58 |
| 59 |         | 59 |
| 60 |         | 60 |

|          |        |          |                                         |
|----------|--------|----------|-----------------------------------------|
|          | BANK   | 21       |                                         |
|          | SETLOC | R11      |                                         |
|          | BANK   |          |                                         |
|          | EBANK= | DVCNTR   |                                         |
|          | COUNT* | \$\$/R11 |                                         |
| R10,R11  | CS     | FLAGWRD7 | # IS SERVICER STILL RUNNING?            |
|          | MASK   | AVEGFBIT |                                         |
|          | CCS    | A        |                                         |
|          | TCF    | TASKOVER | # LET AVGEND TAKE CARE OF GROUP 2.      |
|          | CCS    | PIPCTR   |                                         |
|          | TCF    | +2       |                                         |
|          | TCF    | LRHTASK  | # LAST PASS. CALL LRHTASK.              |
| +2       | TS     | PIPCTR1  |                                         |
| PIPCTR1  | =      | LADQSAVE |                                         |
| PIPCTR   | =      | PHSPRDT2 |                                         |
|          | CAF    | OCT31    |                                         |
|          | TC     | TWIDDLE  |                                         |
| R10,R11A | ADRES  | R10,R11  |                                         |
|          | CS     | IMODES33 | # IF LAMP TEST, DO NOT CHANGE LR LITES. |
|          | MASK   | BIT1     |                                         |
|          | EXTEND |          |                                         |
|          | BZF    | 10,11    |                                         |
| FLASHH?  | MASK   | FLGWRD11 | # C(A) = 1 = HFLASH BIT                 |
|          | EXTEND |          |                                         |
|          | BZF    | FLASHV?  | # H FLASH OFF, SO LEAVE ALONE           |
|          | CA     | HLITE    |                                         |
|          | TS     | L        |                                         |
|          | TC     | FLIP     | # FLIP H LITE                           |
| FLASHV?  | CA     | VFLSHBIT | # VLASHBIT MUST BE BIT 2.               |
|          | MASK   | FLGWRD11 |                                         |
|          | EXTEND |          |                                         |
|          | BZF    | 10,11    | # V FLASH OFF                           |
|          | CA     | VLITE    |                                         |
|          | TS     | L        |                                         |
|          | TC     | FLIP     | # FLIP V LITE                           |
| 10,11    | CA     | FLAGWRD9 | # IS THE LETABORT FLAG SET ?            |
|          | MASK   | LETABBIT |                                         |
|          | EXTEND |          |                                         |
|          | BZF    | LANDISP  | # NO. PROCEED TO R10.                   |
| P71NOW?  | CS     | MODREG   | # YES. ARE WE IN P71 NOW?               |

|    |          |        |          |                                    |    |
|----|----------|--------|----------|------------------------------------|----|
| 1  |          |        |          |                                    | 1  |
| 2  |          | AD     | 1DEC71   |                                    | 2  |
| 3  |          | EXTEND |          |                                    | 3  |
| 4  |          | BZF    | LANDISP  | # YES. PROCEED TO R10.             | 4  |
| 5  |          |        |          |                                    | 5  |
| 6  |          | EXTEND |          | # NO. IS AN ABORT STAGE COMMANDED? | 6  |
| 7  |          | READ   | CHAN30   |                                    | 7  |
| 8  |          | COM    |          |                                    | 8  |
| 9  |          | TS     | L        |                                    | 9  |
| 10 |          | MASK   | BIT4     |                                    | 10 |
| 11 |          | CCS    | A        |                                    | 11 |
| 12 |          | TCF    | P71A     | # YES.                             | 12 |
| 13 |          |        |          |                                    | 13 |
| 14 | P70NOW?  | CS     | MODREG   | # NO. ARE WE IN P70 NOW?           | 14 |
| 15 |          | AD     | 1DEC70   |                                    | 15 |
| 16 |          | EXTEND |          |                                    | 16 |
| 17 |          | BZF    | LANDISP  | # YES. PROCEED TO R10.             | 17 |
| 18 |          |        |          |                                    | 18 |
| 19 |          | CA     | L        | # NO. IS AN ABORT COMMANDED?       | 19 |
| 20 |          | MASK   | BIT1     |                                    | 20 |
| 21 |          | CCS    | A        |                                    | 21 |
| 22 |          | TCF    | P70A     | # YES.                             | 22 |
| 23 |          | TCF    | LANDISP  | # NO. PROCEED TO R10.              | 23 |
| 24 |          |        |          |                                    | 24 |
| 25 |          | COUNT* | \$\$/P70 |                                    | 25 |
| 26 |          |        |          |                                    | 26 |
| 27 | P70      | TC     | LEGAL?   |                                    | 27 |
| 28 | P70A     | CS     | ZERO     |                                    | 28 |
| 29 |          | TCF    | +3       |                                    | 29 |
| 30 | P71      | TC     | LEGAL?   |                                    | 30 |
| 31 | P71A     | CAF    | TWO      |                                    | 31 |
| 32 | +3       | TS     | Q        |                                    | 32 |
| 33 |          | INHINT |          |                                    | 33 |
| 34 |          | EXTEND |          |                                    | 34 |
| 35 |          | DCA    | CNTABTAD |                                    | 35 |
| 36 |          | DTCB   |          |                                    | 36 |
| 37 |          |        |          |                                    | 37 |
| 38 |          | EBANK= | DVCNTR   |                                    | 38 |
| 39 | CNTABTAD | 2CADR  | CONTABRT |                                    | 39 |
| 40 |          |        |          |                                    | 40 |
| 41 | 1DEC70   | DEC    | 70       |                                    | 41 |
| 42 | 1DEC71   | DEC    | 71       |                                    | 42 |
| 43 |          |        |          |                                    | 43 |
| 44 |          | BANK   | 05       |                                    | 44 |
| 45 |          | SETLOC | ABORTS1  |                                    | 45 |
| 46 |          | BANK   |          |                                    | 46 |
| 47 |          | COUNT* | \$\$/P70 |                                    | 47 |
| 48 |          |        |          |                                    | 48 |
| 49 | CONTABRT | CAF    | ABRTJADR |                                    | 49 |
| 50 |          | TS     | BRUPT    |                                    | 50 |
| 51 |          | RESUME |          |                                    | 51 |
| 52 |          |        |          |                                    | 52 |
| 53 |          |        |          |                                    | 53 |
| 54 |          |        |          |                                    | 54 |
| 55 |          |        |          |                                    | 55 |
| 56 |          |        |          |                                    | 56 |
| 57 |          |        |          |                                    | 57 |
| 58 |          |        |          |                                    | 58 |
| 59 |          |        |          |                                    | 59 |
| 60 |          |        |          |                                    | 60 |

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| ABRTJADR | TCF    | ABRTJASK |                                            |
| ABRTJASK | CAF    | OCTAL27  |                                            |
|          | AD     | Q        |                                            |
|          | TS     | L        |                                            |
|          | COM    |          |                                            |
|          | DXCH   | -PHASE4  |                                            |
|          | INDEX  | Q        |                                            |
|          | CAF    | MODE70   |                                            |
|          | TS     | MODREG   |                                            |
|          | TS     | DISPDEX  | # INSURE DISPDEX IS POSITIVE.              |
|          | CCS    | Q        | # SET APSFLAG IF P71.                      |
|          | CS     | FLGWRD10 | # SET APSFLAG PRIOR TO THE ENEMA.          |
|          | MASK   | APSFLBIT |                                            |
|          | ADS    | FLGWRD10 |                                            |
|          | CS     | DAPBITS  | # DAPBITS = OCT 640 = BITS 6, 8, 9         |
|          | MASK   | DAPBOOLS | # (TURN OFF: ULLAGE, DRIFT, AND XOVINHIB ) |
|          | TS     | DAPBOOLS |                                            |
|          | CS     | FLAGWRD5 | # SET ENGONFLG.                            |
|          | MASK   | ENGONBIT |                                            |
|          | ADS    | FLAGWRD5 |                                            |
|          | CS     | PRI030   | # INSURE THAT THE ENGINE IS ON, IF ARMED.  |
|          | EXTEND |          |                                            |
|          | RAND   | DSALMOUT |                                            |
|          | AD     | BIT13    |                                            |
|          | EXTEND |          |                                            |
|          | WRITE  | DSALMOUT |                                            |
|          | CAF    | LRBYBIT  | # TERMINATE R12.                           |
|          | TS     | FLGWRD11 |                                            |
|          | CS     | FLAGWRD0 | # SET R10FLAG TO SUPPRESS OUTPUTS TO THE   |
|          | MASK   | R10FLBIT | # CROSS-POINTER DISPLAY.                   |
|          | ADS    | FLAGWRD0 | # THE FOLLOWING ENEMA WILL REMOVE THE      |
|          |        |          | # DISPLAY INERTIAL DATA OUTBIT.            |
|          | TC     | CLRADM0D | # INSURE RADMODES PROPERLY SET FOR R29.    |
|          | EXTEND |          | # LOAD TEVENT FOR THE DOWNLINK.            |
|          | DCA    | TIME2    |                                            |
|          | DXCH   | TEVENT   |                                            |
|          | EXTEND |          |                                            |
|          | DCA    | SVEXITAD |                                            |
|          | DXCH   | AVGEXIT  |                                            |

|    |          |        |                                  |    |
|----|----------|--------|----------------------------------|----|
| 1  |          |        |                                  | 1  |
| 2  |          |        |                                  | 2  |
| 3  |          | EXTEND |                                  | 3  |
| 4  |          | DCA    | NEGO                             | 4  |
| 5  |          | DXCH   | -PHASE1                          | 5  |
| 6  |          | EXTEND |                                  | 6  |
| 7  |          | DCA    | NEGO                             | 7  |
| 8  |          | DXCH   | -PHASE3                          | 8  |
| 9  |          |        |                                  | 9  |
| 10 |          | EXTEND |                                  | 10 |
| 11 |          | DCA    | NEGO                             | 11 |
| 12 |          | DXCH   | -PHASE6                          | 12 |
| 13 |          |        |                                  | 13 |
| 14 |          | CAF    | THREE                            | 14 |
| 15 |          | TS     | L                                | 15 |
| 16 |          |        | # SET UP 4.3SPOT FOR GOABORT     | 16 |
| 17 |          | COM    |                                  | 17 |
| 18 |          | DXCH   | -PHASE4                          | 18 |
| 19 |          | CAF    | OCT37774                         | 19 |
| 20 |          | TS     | TIME5                            | 20 |
| 21 |          |        | # SET T5RUPT TO CALL DAPIDLER IN | 21 |
| 22 |          |        | # 40 MILLISECONDS.               | 22 |
| 23 |          | TC     | POSTJUMP                         | 23 |
| 24 |          | CADR   | ENEMA                            | 24 |
| 25 |          |        |                                  | 25 |
| 26 | SVEXITAD | EBANK= | DVCNTR                           | 26 |
| 27 |          | 2CADR  | SERVEXIT                         | 27 |
| 28 | MODE70   | DEC    | 70                               | 28 |
| 29 | OCTAL27  | OCT    | 27                               | 29 |
| 30 | MODE71   | DEC    | 71                               | 30 |
| 31 |          |        |                                  | 31 |
| 32 | DAPBITS  | OCT    | 00640                            | 32 |
| 33 |          |        |                                  | 33 |
| 34 |          | BANK   | 32                               | 34 |
| 35 |          | SETLOC | ABORTS                           | 35 |
| 36 |          | BANK   |                                  | 36 |
| 37 |          |        |                                  | 37 |
| 38 |          | COUNT* | \$\$/P70                         | 38 |
| 39 |          |        |                                  | 39 |
| 40 | GOABORT  | TC     | INTPRET                          | 40 |
| 41 |          | CALL   |                                  | 41 |
| 42 |          |        | INITCDUW                         | 42 |
| 43 |          | EXIT   |                                  | 43 |
| 44 |          | CAF    | FOUR                             | 44 |
| 45 |          | TS     | DVCNTR                           | 45 |
| 46 |          |        |                                  | 46 |
| 47 |          | CAF    | WHICHADR                         | 47 |
| 48 |          | TS     | WHICH                            | 48 |
| 49 |          |        |                                  | 49 |
| 50 |          | TC     | DOWNFLAG                         | 50 |
| 51 |          | ADRES  | FLRCS                            | 51 |
| 52 |          |        |                                  | 52 |
| 53 |          |        |                                  | 53 |
| 54 |          |        |                                  | 54 |
| 55 |          |        |                                  | 55 |
| 56 |          |        |                                  | 56 |
| 57 |          |        |                                  | 57 |
| 58 |          |        |                                  | 58 |
| 59 |          |        |                                  | 59 |
| 60 |          |        |                                  | 60 |

|    |           |        |                                            |    |
|----|-----------|--------|--------------------------------------------|----|
| 1  | # P70 P71 |        |                                            | 1  |
| 2  |           | TC     | DOWNFLAG                                   | 2  |
| 3  |           | ADRES  | FLUNDISP                                   | 3  |
| 4  |           |        |                                            | 4  |
| 5  |           | TC     | DOWNFLAG                                   | 5  |
| 6  |           | ADRES  | IDLEFLAG                                   | 6  |
| 7  |           |        |                                            | 7  |
| 8  |           | TC     | UPFLAG                                     | 8  |
| 9  |           | ADRES  | ACC4-2FL                                   | 9  |
| 10 |           |        | # INSURE 4-JET TRANSLATION CAPABILITY.     | 10 |
| 11 |           |        |                                            | 11 |
| 12 | 70DEC     | TC     | CHECKMM                                    | 12 |
| 13 |           | DEC    | 70                                         | 13 |
| 14 |           | TCF    | P71RET                                     | 14 |
| 15 | P70INIT   | TC     | INTPRET                                    | 15 |
| 16 |           | CALL   |                                            | 16 |
| 17 |           |        | TGOCOMP                                    | 17 |
| 18 |           | DLOAD  | SL                                         | 18 |
| 19 |           |        | MDOTDPS                                    | 19 |
| 20 |           |        | 4D                                         | 20 |
| 21 |           | BDDV   |                                            | 21 |
| 22 |           |        | MASS                                       | 22 |
| 23 |           | STODL  | TBUP                                       | 23 |
| 24 |           |        | MASS                                       | 24 |
| 25 |           | DDV    | SR1                                        | 25 |
| 26 |           |        | K(1/DV)                                    | 26 |
| 27 |           | STORE  | 1/DV1                                      | 27 |
| 28 |           | STORE  | 1/DV2                                      | 28 |
| 29 |           | STORE  | 1/DV3                                      | 29 |
| 30 |           | BDDV   |                                            | 30 |
| 31 |           |        | K(AT)                                      | 31 |
| 32 |           | STODL  | AT                                         | 32 |
| 33 |           |        | DTDECAY                                    | 33 |
| 34 |           | DCOMP  | SL                                         | 34 |
| 35 |           |        | 11D                                        | 35 |
| 36 |           | STORE  | TTO                                        | 36 |
| 37 |           | SLOAD  | DCOMP                                      | 37 |
| 38 |           |        | DPSVEX                                     | 38 |
| 39 |           | SR2    |                                            | 39 |
| 40 |           | STORE  | VE                                         | 40 |
| 41 |           | SET    | CALL                                       | 41 |
| 42 |           |        | FLAP                                       | 42 |
| 43 |           |        | COMMINIT                                   | 43 |
| 44 |           | AXC,1  | GOTO                                       | 44 |
| 45 |           |        | OD                                         | 45 |
| 46 |           |        | BOTHPOLY                                   | 46 |
| 47 | INJTARG   | AXC,1  |                                            | 47 |
| 48 |           |        | # RETURN HERE IN P71, SET X1 FOR APS COEFF | 48 |
| 49 | BOTHPOLY  | DLOAD* | 8D                                         | 49 |
| 50 |           |        | # TGO D                                    | 50 |
| 51 |           |        | DMP                                        | 51 |
| 52 |           |        | ABTCOF,1                                   | 52 |
| 53 |           |        | TGO                                        | 53 |
| 54 |           |        |                                            | 54 |
| 55 |           |        |                                            | 55 |
| 56 |           |        |                                            | 56 |
| 57 |           |        |                                            | 57 |
| 58 |           |        |                                            | 58 |
| 59 |           |        |                                            | 59 |
| 60 |           |        |                                            | 60 |



|    |          |        |             |                                            |
|----|----------|--------|-------------|--------------------------------------------|
| 1  |          |        |             |                                            |
| 2  |          | DAD*   | DMP         |                                            |
| 3  |          |        | ABTCOF +2,1 | # TGO(C+TGO D)                             |
| 4  |          |        | TGO         |                                            |
| 5  |          | DAD*   | DMP         |                                            |
| 6  |          |        | ABTCOF +4,1 | # TGO(B+TGO(C+TGO D))                      |
| 7  |          |        | TGO         |                                            |
| 8  |          | DAD*   |             |                                            |
| 9  |          |        | ABTCOF +6,1 | # A+TGO(B+TGO(C+TGO D))                    |
| 10 |          | STORE  | ZDOTD       | # STORE TENTATIVELY IN ZDOTD               |
| 11 |          | DSU    | BPL         | # CHECK AGAINST MINIMUM                    |
| 12 |          |        | VMIN        |                                            |
| 13 |          | DLOAD  | UPRATE      | # IF BIG ENOUGH, LEAVE ZDOTD AS IS .       |
| 14 |          |        | VMIN        |                                            |
| 15 |          |        |             |                                            |
| 16 |          | STORE  | ZDOTD       | # IF TOO SMALL, REPLACE WITH MINIMUM.      |
| 17 | UPRATE   | DLOAD  |             |                                            |
| 18 |          |        | ABTRDOT     |                                            |
| 19 |          | STCALL | RDOTD       | # INITIALZE RDOTD.                         |
| 20 |          |        | YCOMP       | # COMPUTE Y                                |
| 21 |          | ABS    | DSU         |                                            |
| 22 |          |        | YLIM        | # /Y/-DYMAX                                |
| 23 |          | BMN    | SIGN        | # IF <0, XR<.5DEG, LEAVE YCO AT 0          |
| 24 |          |        | YOK         | # IF >0, FIX SIGN OF DEFICIT, THIS IS YCO. |
| 25 |          |        | Y           |                                            |
| 26 |          | STORE  | YCO         |                                            |
| 27 | YOK      | DLOAD  | DSU         |                                            |
| 28 |          |        | YCO         |                                            |
| 29 |          |        | Y           | # COMPUTE X RANGE IN CASE ASTRONAUT WANTS  |
| 30 |          | SR     |             |                                            |
| 31 |          |        | 5D          |                                            |
| 32 |          | STORE  | XRANGE      | # TO LOOK.                                 |
| 33 | UPTHROT  | SET    | EXIT        |                                            |
| 34 |          |        | FLVR        |                                            |
| 35 |          |        |             |                                            |
| 36 |          | TC     | UPFLAG      | # SET ROTFLAG                              |
| 37 |          | ADRES  | ROTFLAG     |                                            |
| 38 |          |        |             |                                            |
| 39 |          | TC     | THROTUP     |                                            |
| 40 |          |        |             |                                            |
| 41 |          | TC     | PHASCHNG    |                                            |
| 42 |          | OCT    | 04024       |                                            |
| 43 |          |        |             |                                            |
| 44 | -3       | TC     | BANKCALL    | # VERIFY THAT THE PANEL SWITCHES           |
| 45 |          | CADR   | P40AUTO     | # ARE PROPERLY SET.                        |
| 46 |          |        |             |                                            |
| 47 |          | TC     | THROTUP     |                                            |
| 48 |          |        |             |                                            |
| 49 | UPTHROT1 | EXTEND |             | # SET SERVICER TO CALL ASCENT GUIDANCE.    |
| 50 |          | DCA    | ATMAGAD     |                                            |
| 51 |          | DXCH   | AVGEXIT     |                                            |
| 52 |          |        |             |                                            |
| 53 |          |        |             |                                            |
| 54 |          |        |             |                                            |
| 55 |          |        |             |                                            |
| 56 |          |        |             |                                            |
| 57 |          |        |             |                                            |
| 58 |          |        |             |                                            |
| 59 |          |        |             |                                            |
| 60 |          |        |             |                                            |

|         |                        |                             |                                            |
|---------|------------------------|-----------------------------|--------------------------------------------|
| GRP4OFF | TC<br>OCT              | PHASCHNG<br>00004           | # TERMINATE USE OF GROUP 4.                |
|         | TCF                    | ENDOFJOB                    |                                            |
| P71RET  | TC<br>ADRES            | DOWNFLAG<br>LETABORT        |                                            |
|         | CAF<br>TS              | THRESH2<br>DVTHRUSH         | # SET DVMON THRESHOLD TO THE ASCENT VALUE. |
|         | TC<br>BON              | INTPRET<br>CALL<br>FLAP     |                                            |
|         | SSP                    | OLDTIME<br>TGOCOMP<br>GOTO  | # IF FLAP=0, TGO=T-TIG                     |
|         | CADR                   | QPRET<br>INJTARG<br>P12INIT | # WILL EXIT P12INIT TO INJTARG             |
| OLDTIME | DLOAD                  | SL1<br>TGO                  | # IF FLAP=1,GTO=2 TGO                      |
|         | STCALL                 | TGO1<br>P12INIT             |                                            |
|         | EXIT<br>TC<br>OCT      | PHASCHNG<br>04024           |                                            |
|         | EXTEND                 |                             |                                            |
|         | DCA<br>DXCH<br>TCF     | TGO1<br>TGO<br>UPTHROT1 -3  |                                            |
| TGO1    | =                      | VGBODY                      |                                            |
| # ***** |                        |                             |                                            |
|         | BANK<br>SETLOC<br>BANK | 21<br>R11                   |                                            |
|         | COUNT*                 | \$\$/P70                    |                                            |
| LEGAL?  | CS<br>AD               | MMNUMBER<br>MODREG          | # IS THE DESIRED PGM ALREADY IN PROGRESS?  |
|         | EXTEND<br>BZF          | ABORTALM                    |                                            |
|         | CS<br>MASK<br>CCS      | FLAGWRD9<br>LETABBIT<br>A   | # ARE THE ABORTS ENABLED?                  |

|    |          |        |                                               |    |
|----|----------|--------|-----------------------------------------------|----|
| 1  |          |        |                                               | 1  |
| 2  |          | TCF    | ABORTALM                                      | 2  |
| 3  |          |        |                                               | 3  |
| 4  |          | CA     | FLAGWRD7 # IS SERVICER ON THE AIR?            | 4  |
| 5  |          | MASK   | AVEGFBIT                                      | 5  |
| 6  |          | CCS    | A                                             | 6  |
| 7  | ABORTALM | TC     | Q # YES. ALL IS WELL.                         | 7  |
| 8  |          | TC     | FALTON                                        | 8  |
| 9  |          | TC     | RELDSP                                        | 9  |
| 10 |          | TC     | POSTJUMP                                      | 10 |
| 11 |          | CADR   | PINBRNCH                                      | 11 |
| 12 |          |        |                                               | 12 |
| 13 |          | BANK   | 32                                            | 13 |
| 14 |          | SETLOC | ABORTS                                        | 14 |
| 15 |          | BANK   |                                               | 15 |
| 16 |          |        |                                               | 16 |
| 17 |          | COUNT* | \$\$/P70                                      | 17 |
| 18 |          |        |                                               | 18 |
| 19 | # *****  |        |                                               | 19 |
| 20 |          |        |                                               | 20 |
| 21 | TGOCOMP  | RTB    | DSU                                           | 21 |
| 22 |          |        | LOADTIME                                      | 22 |
| 23 |          |        | TIG                                           | 23 |
| 24 |          | SL     |                                               | 24 |
| 25 |          |        | 11D                                           | 25 |
| 26 |          | STORE  | TGO                                           | 26 |
| 27 |          | RVQ    |                                               | 27 |
| 28 |          |        |                                               | 28 |
| 29 | # *****  |        |                                               | 29 |
| 30 |          |        |                                               | 30 |
| 31 | THROTUP  | CAF    | BIT13                                         | 31 |
| 32 |          | TS     | THRUST                                        | 32 |
| 33 |          | CAF    | BIT4                                          | 33 |
| 34 |          | EXTEND |                                               | 34 |
| 35 |          | WOR    | CHAN14                                        | 35 |
| 36 |          | TC     | Q                                             | 36 |
| 37 |          |        |                                               | 37 |
| 38 | # *****  |        |                                               | 38 |
| 39 |          |        |                                               | 39 |
| 40 | 10SECS   | 2DEC   | 1000                                          | 40 |
| 41 |          |        |                                               | 41 |
| 42 | HINJECT  | 2DEC   | 18288 B-24 # 60,000 FEET EXPRESSED IN METERS. | 42 |
| 43 |          |        |                                               | 43 |
| 44 | (TGO)A   | 2DEC   | 37000 B-17                                    | 44 |
| 45 |          |        |                                               | 45 |
| 46 | K(AT)    | 2DEC   | .02 # SCALING CONSTANT                        | 46 |
| 47 |          |        |                                               | 47 |
| 48 | WHICHADR | REMADR | ABRTABLE                                      | 48 |
| 49 |          |        |                                               | 49 |
| 50 | # *****  |        |                                               | 50 |
| 51 |          |        |                                               | 51 |
| 52 |          |        |                                               | 52 |
| 53 |          |        |                                               | 53 |
| 54 |          |        |                                               | 54 |
| 55 |          |        |                                               | 55 |
| 56 |          |        |                                               | 56 |
| 57 |          |        |                                               | 57 |
| 58 |          |        |                                               | 58 |
| 59 |          |        |                                               | 59 |
| 60 |          |        |                                               | 60 |



|    |          |                 |                 |    |
|----|----------|-----------------|-----------------|----|
| 1  |          |                 |                 | 1  |
| 2  |          |                 |                 | 2  |
| 3  | ATMAGAD  | EBANK=<br>2CADR | DVCNTR<br>ATMAG | 3  |
| 4  |          |                 |                 | 4  |
| 5  | ORBMANAD | ADRES           | ORBMANUV        | 5  |
| 6  |          |                 |                 | 6  |
| 7  |          |                 |                 | 7  |
| 8  |          |                 |                 | 8  |
| 9  |          |                 |                 | 9  |
| 10 |          |                 |                 | 10 |
| 11 |          |                 |                 | 11 |
| 12 |          |                 |                 | 12 |
| 13 |          |                 |                 | 13 |
| 14 |          |                 |                 | 14 |
| 15 |          |                 |                 | 15 |
| 16 |          |                 |                 | 16 |
| 17 |          |                 |                 | 17 |
| 18 |          |                 |                 | 18 |
| 19 |          |                 |                 | 19 |
| 20 |          |                 |                 | 20 |
| 21 |          |                 |                 | 21 |
| 22 |          |                 |                 | 22 |
| 23 |          |                 |                 | 23 |
| 24 |          |                 |                 | 24 |
| 25 |          |                 |                 | 25 |
| 26 |          |                 |                 | 26 |
| 27 |          |                 |                 | 27 |
| 28 |          |                 |                 | 28 |
| 29 |          |                 |                 | 29 |
| 30 |          |                 |                 | 30 |
| 31 |          |                 |                 | 31 |
| 32 |          |                 |                 | 32 |
| 33 |          |                 |                 | 33 |
| 34 |          |                 |                 | 34 |
| 35 |          |                 |                 | 35 |
| 36 |          |                 |                 | 36 |
| 37 |          |                 |                 | 37 |
| 38 |          |                 |                 | 38 |
| 39 |          |                 |                 | 39 |
| 40 |          |                 |                 | 40 |
| 41 |          |                 |                 | 41 |
| 42 |          |                 |                 | 42 |
| 43 |          |                 |                 | 43 |
| 44 |          |                 |                 | 44 |
| 45 |          |                 |                 | 45 |
| 46 |          |                 |                 | 46 |
| 47 |          |                 |                 | 47 |
| 48 |          |                 |                 | 48 |
| 49 |          |                 |                 | 49 |
| 50 |          |                 |                 | 50 |
| 51 |          |                 |                 | 51 |
| 52 |          |                 |                 | 52 |
| 53 |          |                 |                 | 53 |
| 54 |          |                 |                 | 54 |
| 55 |          |                 |                 | 55 |
| 56 |          |                 |                 | 56 |
| 57 |          |                 |                 | 57 |
| 58 |          |                 |                 | 58 |
| 59 |          |                 |                 | 59 |
| 60 |          |                 |                 | 60 |

|       |        |          |                                            |
|-------|--------|----------|--------------------------------------------|
|       | BANK   | 24       |                                            |
|       | SETLOC | P12      |                                            |
|       | BANK   |          |                                            |
|       | EBANK= | DVCNTR   |                                            |
|       | COUNT* | \$\$/P12 |                                            |
| P12LM | TC     | PHASCHNG |                                            |
|       | OCT    | 04024    |                                            |
|       | TC     | BANKCALL |                                            |
|       | CADR   | R02BOTH  | # CHECK THE STATUS OF THE IMU.             |
|       | TC     | UPFLAG   |                                            |
|       | ADRES  | MUNFLAG  |                                            |
|       | TC     | UPFLAG   | # INSURE 4-JET TRANSLATION CAPABILITY.     |
|       | ADRES  | ACC4-2FL |                                            |
|       | TC     | UPFLAG   | # PREVENT R10 FROM ISSUING CROSS-POINTER   |
|       | ADRES  | R10FLAG  | # OUTPUTS.                                 |
|       | TC     | CLRADMOD | # INITIALIZE RADMODES FOR R29.             |
|       | TC     | DOWNFLAG | # CLEAR RENDEZVOUS FLAG FOR P22            |
|       | ADRES  | RNDVZFLG |                                            |
|       | CAF    | THRESH2  | # INITIALIZE DVMON                         |
|       | TS     | DVTHRUSH |                                            |
|       | CAF    | FOUR     |                                            |
|       | TS     | DVCNTR   |                                            |
|       | CA     | ZERO     |                                            |
|       | TS     | TRKMKCNT | # SHOW THAT R29 DOWNLINK DATA ISN'T READY. |
|       | CAF    | V06N33A  |                                            |
|       | TC     | BANKCALL | # FLASH TIG                                |
|       | CADR   | GOFLASH  |                                            |
|       | TCF    | GOTOP00H |                                            |
|       | TCF    | +2       | # PROCEED                                  |
|       | TCF    | -5       | # ENTER                                    |
|       | TC     | PHASCHNG |                                            |
|       | OCT    | 04024    |                                            |
|       | TC     | INTPRET  |                                            |
|       | CALL   |          | # INITIALZE WM AND /LAND/                  |
|       |        | GUIDINIT |                                            |
|       | SET    | CALL     |                                            |
|       |        | FLPI     |                                            |
|       |        | P12INIT  |                                            |

|    |         |        |          |                                          |
|----|---------|--------|----------|------------------------------------------|
| 1  | P12LMB  | DLOAD  | (TGO)A   | # SET TGO TO AN INITIAL NOMINAL VALUE.   |
| 2  |         | STODL  | TGO      |                                          |
| 3  |         |        | TIG      |                                          |
| 4  |         | STCALL | TDEC1    |                                          |
| 5  |         | VLOAD  | LEMPREC  | # ROTATE THE STATE VECTORS TO THE        |
| 6  |         |        | MXV      | # IGNITION TIME.                         |
| 7  |         |        | VATT     |                                          |
| 8  |         |        | REFSMMAT |                                          |
| 9  |         | VSL1   |          |                                          |
| 10 |         | STOVL  | V1S      | # COMPUTE V1S = VEL(TIG)*2(-7)M/CS.      |
| 11 |         | MXV    | RATT     |                                          |
| 12 |         |        | VSL6     |                                          |
| 13 |         |        | REFSMMAT |                                          |
| 14 |         | STCALL | R        | # COMPUTE R = POS(TIG)*2(-24)M.          |
| 15 |         |        | MUNGRAV  | # COMPUTE GDT1/2(TIG)*2(-7)M/CS.         |
| 16 |         | VLOAD  | UNIT     |                                          |
| 17 |         |        | R        |                                          |
| 18 |         | STCALL | UNIT/R/  | # COMPUTE UNIT/R/ FOR YCOMP.             |
| 19 |         |        | YCOMP    |                                          |
| 20 |         | SR     | DCOMP    |                                          |
| 21 |         |        | 5D       |                                          |
| 22 |         | STODL  | XRANGE   | # INITIALIZE XRANGE FOR NOUN 76.         |
| 23 |         |        | VINJNOM  |                                          |
| 24 |         | STODL  | ZDOTD    |                                          |
| 25 |         |        | RDOTDNOM |                                          |
| 26 |         | STORE  | RDOTD    |                                          |
| 27 |         | EXIT   |          |                                          |
| 28 |         | TC     | PHASCHNG |                                          |
| 29 |         | OCT    | 04024    |                                          |
| 30 |         |        |          |                                          |
| 31 | NEWLOAD | CAF    | V06N76   | # FLASH CROSS-RANGE, AND APOLUNE VALUES. |
| 32 |         | TC     | BANKCALL |                                          |
| 33 |         | CADR   | GOFLASH  |                                          |
| 34 |         | TCF    | GOTOPOOH |                                          |
| 35 |         | TCF    | +2       | # PROCEED                                |
| 36 |         | TCF    | NEWLOAD  | # ENTER NEW DATA.                        |
| 37 |         |        |          |                                          |
| 38 |         | CAF    | P12ADRES |                                          |
| 39 |         | TS     | WHICH    |                                          |
| 40 |         |        |          |                                          |
| 41 |         | TC     | PHASCHNG |                                          |
| 42 |         | OCT    | 04024    |                                          |
| 43 |         |        |          |                                          |
| 44 |         | TC     | INTPRET  |                                          |
| 45 |         | DLOAD  | SL       |                                          |
| 46 |         |        | XRANGE   |                                          |
| 47 |         |        | 5D       |                                          |
| 48 |         | DAD    |          |                                          |
| 49 |         |        |          |                                          |
| 50 |         |        |          |                                          |
| 51 |         |        |          |                                          |
| 52 |         |        |          |                                          |
| 53 |         |        |          |                                          |
| 54 |         |        |          |                                          |
| 55 |         |        |          |                                          |
| 56 |         |        |          |                                          |
| 57 |         |        |          |                                          |
| 58 |         |        |          |                                          |
| 59 |         |        |          |                                          |
| 60 |         |        |          |                                          |

|    |         |        |                         |                                            |          |                                              |
|----|---------|--------|-------------------------|--------------------------------------------|----------|----------------------------------------------|
|    | # P12   |        |                         |                                            | PAGE 010 |                                              |
| 1  |         |        | Y                       |                                            |          | 1                                            |
| 2  |         | STOVL  | YCO                     |                                            |          | 2                                            |
| 3  |         | VXSC   | UNIT/R/<br>VAD<br>49FPS |                                            |          | 3<br>4<br>5<br>6<br>7<br>8                   |
| 4  |         | STORE  | VIS                     |                                            |          | 9                                            |
| 5  |         | DOT    | V                       | # V(TIPOVER) = V(IGN) + 57FPS (UNIT/R/)    |          | 10                                           |
| 6  |         |        | SL1                     |                                            |          | 11                                           |
| 7  |         | STOVL  | UNIT/R/<br>RDOT         | # RDOT * 2(-7)                             |          | 12<br>13<br>14<br>15<br>16                   |
| 8  |         | VXV    | UNIT                    |                                            |          | 17                                           |
| 9  |         | STORE  | QAXIS                   |                                            |          | 18                                           |
| 10 |         | SETGO  | ZAXIS1                  |                                            |          | 19<br>20                                     |
| 11 |         |        | FLVR                    |                                            |          | 21                                           |
| 12 |         |        | ASCENT                  |                                            |          | 22<br>23<br>24                               |
| 13 | P12RET  | DLOAD  | ATP                     | # ATP(2)*2(18)                             |          | 25                                           |
| 14 |         | DSQ    | PDDL                    |                                            |          | 26                                           |
| 15 |         | DSQ    | ATY                     | # ATY(2)*2(18)                             |          | 27                                           |
| 16 |         | BZE    | DAD                     |                                            |          | 28                                           |
| 17 |         |        | SQRT                    |                                            |          | 29                                           |
| 18 |         | SL1    | YAWDUN                  |                                            |          | 30                                           |
| 19 |         |        | BDDV                    |                                            |          | 31                                           |
| 20 |         |        | ATY                     |                                            |          | 32                                           |
| 21 | YAWDUN  | ARCSIN |                         |                                            |          | 33                                           |
| 22 |         | STOVL  | YAW                     |                                            |          | 34                                           |
| 23 |         |        | UNFC/2                  |                                            |          | 35                                           |
| 24 |         | UNIT   | DOT                     |                                            |          | 36                                           |
| 25 |         | SL1    | UNIT/R/<br>ARCCOS       |                                            |          | 37<br>38<br>39<br>40                         |
| 26 |         | DCOMP  |                         |                                            |          | 41                                           |
| 27 |         | STORE  | PITCH                   |                                            |          | 42                                           |
| 28 |         | EXIT   |                         |                                            |          | 43                                           |
| 29 |         | TC     | PHASCHNG                |                                            |          | 44                                           |
| 30 |         | OCT    | 04024                   |                                            |          | 45<br>46<br>47<br>48                         |
| 31 |         | TC     | DOWNFLAG                |                                            |          | 49                                           |
| 32 |         | ADRES  | FLPI                    |                                            |          | 50<br>51<br>52                               |
| 33 |         | INHINT |                         |                                            |          | 53                                           |
| 34 |         | TC     | IBNKCALL                |                                            |          | 54                                           |
| 35 |         | CADR   | PFLITEDB                |                                            |          | 55<br>56<br>57                               |
| 36 |         | RELINT |                         |                                            |          | 58                                           |
| 37 |         | TC     | POSTJUMP                |                                            |          | 59                                           |
| 38 |         | CADR   | BURNBABY                |                                            |          | 60<br>61<br>62<br>63<br>64                   |
| 39 | P12INIT | DLOAD  |                         | # INITIALIZE ENGINE DATA. USED FOR P12 AND |          | 65<br>66<br>67<br>68<br>69<br>70<br>71<br>72 |
| 40 |         |        |                         |                                            |          | 73<br>74<br>75<br>76<br>77<br>78<br>79<br>80 |







|    |          |        |             |                                          |    |
|----|----------|--------|-------------|------------------------------------------|----|
| 1  |          |        |             |                                          | 1  |
| 2  |          | MXV    | VXSC        |                                          | 2  |
| 3  |          |        | REFSMMAT    |                                          | 3  |
| 4  |          |        | MOONRATE    |                                          | 4  |
| 5  |          | STOVL  | WM          |                                          | 5  |
| 6  |          |        | RLS         |                                          | 6  |
| 7  |          | ABVAL  | SL3         |                                          | 7  |
| 8  |          | STCALL | /LAND/      |                                          | 8  |
| 9  |          |        | TEMPR60     |                                          | 9  |
| 10 |          |        |             |                                          | 10 |
| 11 | 49FPS    | 2DEC   | .149352 B-6 | # EXPECTED RDOT AT TIPOVER               | 11 |
| 12 | VINJNOM  | 2DEC   | 16.7924 B-7 | # 5509.5 FPS(APO=30NM WITH RDOT=19.5FPS) | 12 |
| 13 | RDOTDNOM | 2DEC   | .059436 B-7 | # 19.5 FPS                               | 13 |
| 14 |          |        |             |                                          | 14 |
| 15 |          |        |             |                                          | 15 |
| 16 |          |        |             |                                          | 16 |
| 17 |          |        |             |                                          | 17 |
| 18 |          |        |             |                                          | 18 |
| 19 |          |        |             |                                          | 19 |
| 20 |          |        |             |                                          | 20 |
| 21 |          |        |             |                                          | 21 |
| 22 |          |        |             |                                          | 22 |
| 23 |          |        |             |                                          | 23 |
| 24 |          |        |             |                                          | 24 |
| 25 |          |        |             |                                          | 25 |
| 26 |          |        |             |                                          | 26 |
| 27 |          |        |             |                                          | 27 |
| 28 |          |        |             |                                          | 28 |
| 29 |          |        |             |                                          | 29 |
| 30 |          |        |             |                                          | 30 |
| 31 |          |        |             |                                          | 31 |
| 32 |          |        |             |                                          | 32 |
| 33 |          |        |             |                                          | 33 |
| 34 |          |        |             |                                          | 34 |
| 35 |          |        |             |                                          | 35 |
| 36 |          |        |             |                                          | 36 |
| 37 |          |        |             |                                          | 37 |
| 38 |          |        |             |                                          | 38 |
| 39 |          |        |             |                                          | 39 |
| 40 |          |        |             |                                          | 40 |
| 41 |          |        |             |                                          | 41 |
| 42 |          |        |             |                                          | 42 |
| 43 |          |        |             |                                          | 43 |
| 44 |          |        |             |                                          | 44 |
| 45 |          |        |             |                                          | 45 |
| 46 |          |        |             |                                          | 46 |
| 47 |          |        |             |                                          | 47 |
| 48 |          |        |             |                                          | 48 |
| 49 |          |        |             |                                          | 49 |
| 50 |          |        |             |                                          | 50 |
| 51 |          |        |             |                                          | 51 |
| 52 |          |        |             |                                          | 52 |
| 53 |          |        |             |                                          | 53 |
| 54 |          |        |             |                                          | 54 |
| 55 |          |        |             |                                          | 55 |
| 56 |          |        |             |                                          | 56 |
| 57 |          |        |             |                                          | 57 |
| 58 |          |        |             |                                          | 58 |
| 59 |          |        |             |                                          | 59 |
| 60 |          |        |             |                                          | 60 |

|    |       |        |            |    |
|----|-------|--------|------------|----|
| 1  |       |        |            | 1  |
| 2  |       | BANK   | 34         | 2  |
| 3  |       | SETLOC | ASCFLT     | 3  |
| 4  |       | BANK   |            | 4  |
| 5  |       |        |            | 5  |
| 6  |       | EBANK= | DVCNTR     | 6  |
| 7  |       |        |            | 7  |
| 8  |       | COUNT* | \$\$/ASENT | 8  |
| 9  |       |        |            | 9  |
| 10 | ATMAG | TC     | PHASCHNG   | 10 |
| 11 |       | OCT    | 00035      | 11 |
| 12 |       | TC     | INTPRET    | 12 |
| 13 |       | BON    |            | 13 |
| 14 |       |        | FLRCS      | 14 |
| 15 |       |        | ASCENT     | 15 |
| 16 |       | DLOAD  | DSU        | 16 |
| 17 |       |        | ABDVCONV   | 17 |
| 18 |       |        | MINABDV    | 18 |
| 19 |       | BMN    | CLEAR      | 19 |
| 20 |       |        | ASCTERM4   | 20 |
| 21 |       |        | SURFFLAG   | 21 |
| 22 |       | CLEAR  | SLOAD      | 22 |
| 23 |       |        | RENDWFLG   | 23 |
| 24 |       |        | BIT3H      | 24 |
| 25 |       | DDV    | EXIT       | 25 |
| 26 |       |        | ABDVCONV   | 26 |
| 27 |       | DXCH   | MPAC       | 27 |
| 28 |       | DXCH   | 1/DV3      | 28 |
| 29 |       | DXCH   | 1/DV2      | 29 |
| 30 |       | DXCH   | 1/DV1      | 30 |
| 31 |       | DXCH   | 1/DV0      | 31 |
| 32 |       | TC     | INTPRET    | 32 |
| 33 |       | DLOAD  | DAD        | 33 |
| 34 |       |        | 1/DV0      | 34 |
| 35 |       |        | 1/DV1      | 35 |
| 36 |       | DAD    | DAD        | 36 |
| 37 |       |        | 1/DV2      | 37 |
| 38 |       |        | 1/DV3      | 38 |
| 39 |       | DMP    | DMP        | 39 |
| 40 |       |        | VE         | 40 |
| 41 |       |        | 2SEC(9)    | 41 |
| 42 |       | SL3    | PDDL       | 42 |
| 43 |       |        | TBUP       | 43 |
| 44 |       | SR1    | DAD        | 44 |
| 45 |       | DSU    |            | 45 |
| 46 |       |        | 6SEC(18)   | 46 |
| 47 |       | STODL  | TBUP       | 47 |
| 48 |       |        | VE         | 48 |
| 49 |       | SR1    | DDV        | 49 |
| 50 |       |        | TBUP       | 50 |
| 51 |       | STCALL | AT         | 51 |
| 52 |       |        |            | 52 |
| 53 |       |        |            | 53 |
| 54 |       |        |            | 54 |
| 55 |       |        |            | 55 |
| 56 |       |        |            | 56 |
| 57 |       |        |            | 57 |
| 58 |       |        |            | 58 |
| 59 |       |        |            | 59 |
| 60 |       |        |            | 60 |

## # ASCENT\_GUIDANCE

## BIT3H

OCT

## ASCENT 4

|        |        |            |                                       |
|--------|--------|------------|---------------------------------------|
|        | BANK   | 30         |                                       |
|        | SETLOC | ASENT      |                                       |
|        | BANK   |            |                                       |
|        | COUNT* | \$\$/ASENT |                                       |
| ASCENT | VLOAD  | ABVAL      |                                       |
|        |        | R          |                                       |
|        | STOVL  | /R/MAG     |                                       |
|        |        | ZAXIS1     |                                       |
|        | DOT    | SL1        |                                       |
|        |        | V          | # Z.V = ZDOT*2(-8).                   |
|        | STOVL  | ZDOT       | # ZDOT*2(-7)                          |
|        |        | ZAXIS1     |                                       |
|        | VXV    | VSL1       |                                       |
|        |        | UNIT/R/    | # Z X UR = LAXIS*2(-2)                |
|        | STORE  | LAXIS      | # LAXIS*2(-1)                         |
|        | DOT    | SL1        |                                       |
|        |        | V          | # L.V = YDOT*2(-8).                   |
|        | STCALL | YDOT       | # YDOT * 2(-7)                        |
|        |        | YCOMP      |                                       |
|        | VLOAD  |            |                                       |
|        |        | GDT1/2     | # LOAD GDT1/2*2(-7) M/CS.             |
|        | V/SC   | DOT        |                                       |
|        |        | 2SEC(18)   |                                       |
|        |        | UNIT/R/    | # G.UR*2(9) = GR*2(9).                |
|        | PDVL   | VXV        | # STORE IN PDL(0)                     |
|        |        | UNIT/R/    | # LOAD UNIT/R/ *2(-1)                 |
|        |        | V          | # UR*2(-1) X V*2(-7) = H/R*2(-8).     |
|        | VSQ    | DDV        | # H(2)/R(2)*2(-16).                   |
|        |        | /R/MAG     | # H(2)/R(3)*2(9).                     |
|        | SL1    | DAD        |                                       |
|        | STADR  |            |                                       |
|        | STODL  | GEFF       | # GEFF*2(10)M/CS/CS.                  |
|        |        | ZDOTD      |                                       |
|        | DSU    |            |                                       |
|        |        | ZDOT       |                                       |
|        | STORE  | DZDOT      | # DZDOT = (ZDOTD - ZDOT) * 2(7) M/CS. |
|        | VXSC   | PDDL       |                                       |
|        |        | ZAXIS1     |                                       |
|        |        | YDOTD      |                                       |
|        | DSU    |            |                                       |
|        |        | YDOT       |                                       |
|        | STORE  | DYDOT      | # DYDOT = (YDOTD - YDOT) *2(7) M/CS.  |
|        | VXSC   | PDDL       |                                       |
|        |        | LAXIS      |                                       |
|        |        | RDOTD      |                                       |

|    |         |        |          |                                        |
|----|---------|--------|----------|----------------------------------------|
| 1  |         | DSU    | RDOT     |                                        |
| 2  |         |        |          |                                        |
| 3  |         | STORE  | DRDOT    | # DRDOT = (RDOTD - RDOT) * 2(7) M/CS.  |
| 4  |         | VXSC   | VAD      |                                        |
| 5  |         |        | UNIT/R/  |                                        |
| 6  |         | VAD    | VSL1     |                                        |
| 7  |         | STADR  |          |                                        |
| 8  |         | STORE  | VGVECT   | # VG = (DRDOT)R + (DVDOT)L + (DZDOT)Z. |
| 9  |         | DLOAD  | DMP      | # LOAD TGO                             |
| 10 |         |        | TGO      | # TGO GEFF                             |
| 11 |         |        | GEFF     |                                        |
| 12 |         | VXSC   | VSL1     |                                        |
| 13 |         |        | UNIT/R/  | # TGO GEFF UR                          |
| 14 |         | BVSU   |          |                                        |
| 15 |         |        | VGVECT   | # COMPENSATED FOR GEFF                 |
| 16 |         | STORE  | VGVECT   | # STORE FOR DOWNLINK                   |
| 17 |         | MXV    | VSL1     | # GET VGBODY FOR N85 DISPLAY           |
| 18 |         |        |          |                                        |
| 19 |         | STOVL  | XNBPIP   |                                        |
| 20 |         |        | VGBODY   |                                        |
| 21 |         |        | VGVECT   |                                        |
| 22 |         | ABVAL  | BOFF     | # MAGNITUDE OF VGVECT                  |
| 23 |         |        | FLRCS    | # IF FLRCS=0,DO NORMAL GUIDANCE        |
| 24 |         |        | MAINENG  |                                        |
| 25 |         | DDV    |          | # USE TGO=VG/AT WITH RCS               |
| 26 |         |        | AT/RCS   |                                        |
| 27 |         | STCALL | TGO      | # THIS WILL BE USED ON NEXT CYCLE      |
| 28 |         |        | ASCTERM2 |                                        |
| 29 | MAINENG | DDV    | PUSH     | # VG/VE IN PDL(0) (2)                  |
| 30 |         |        | VE       |                                        |
| 31 |         | DMP    | BDSU     | # 1 - KT VG/VE                         |
| 32 |         |        | KT1      |                                        |
| 33 |         |        | NEARONE  |                                        |
| 34 |         | DMP    | DMP      | # TBUP VG(1-KT VG/VE)/VE (0)           |
| 35 |         |        | TBUP     | # = TGO                                |
| 36 |         | DSU    |          | # COMPENSATE FOR TAILOFF               |
| 37 |         |        | TTO      |                                        |
| 38 |         | STORE  | TGO      |                                        |
| 39 |         | SR     | DCOMP    |                                        |
| 40 |         |        | 11D      |                                        |
| 41 |         | STODL  | TTOGO    | # TGO *2(-28) CS                       |
| 42 |         |        | TGO      |                                        |
| 43 |         | BON    | DSU      |                                        |
| 44 |         |        | IDLEFLAG |                                        |
| 45 |         |        | T2TEST   |                                        |
| 46 |         |        | 4SEC(17) | # ( TGO - 4 ) *2(-17) CS.              |
| 47 |         | BMN    |          |                                        |
| 48 |         |        | ENGOFF   |                                        |
| 49 | T2TEST  | DLOAD  |          |                                        |
| 50 |         |        | TGO      |                                        |
| 51 |         | DSU    | BMN      | # IF TGO - T2 NEG., GO TO CMPOINT      |

|    |         |       |          |                                      |      |
|----|---------|-------|----------|--------------------------------------|------|
| 1  |         |       |          |                                      |      |
| 2  |         |       | T2A      |                                      |      |
| 3  |         |       | CMponent |                                      |      |
| 4  |         | DLOAD | DSU      |                                      |      |
| 5  |         |       | TBUP     |                                      |      |
| 6  |         |       | TGO      |                                      |      |
| 7  |         | DDV   | CALL     | # 1- TGO/TBUP                        |      |
| 8  |         |       | TBUP     |                                      |      |
| 9  |         |       | LOGSUB   |                                      |      |
| 10 |         | SL    | PUSH     | # -L IN PDL(0)                       | (2)  |
| 11 |         |       | 5        |                                      |      |
| 12 |         | BDDV  | BDSU     | # -TGO/L*2(-17)                      |      |
| 13 |         |       | TGO      |                                      |      |
| 14 |         |       | TBUP     | # TBUP + TGO/L = D12*2(-17)          |      |
| 15 |         | PUSH  | BON      | # STORE IN PDL(2)                    | (4)  |
| 16 |         |       | FLPC     | # IF FLPC = 1, GO TO CONST           |      |
| 17 |         |       | NORATES  |                                      |      |
| 18 |         | DLOAD | DSU      |                                      |      |
| 19 |         |       | TGO      |                                      |      |
| 20 |         |       | T3       |                                      |      |
| 21 |         | BPL   | SET      | # FLPC=1                             |      |
| 22 |         |       | RATES    |                                      |      |
| 23 |         |       | FLPC     |                                      |      |
| 24 | NORATES | DLOAD |          |                                      |      |
| 25 |         |       | HI6ZEROS |                                      |      |
| 26 |         | STORE | PRATE    | # B = 0                              |      |
| 27 |         | STORE | YRATE    | # D = 0                              |      |
| 28 |         | GOTO  |          |                                      |      |
| 29 |         |       | CONST    | # GO TO CONST                        |      |
| 30 | RATES   | DLOAD | DSU      |                                      |      |
| 31 |         |       | TGO      |                                      |      |
| 32 |         |       | 02D      | # TGO - D12 = D21*2(-17)             |      |
| 33 |         | PUSH  | SL1      | # IN PDL(4)                          | (6)  |
| 34 |         | BDSU  | SL3      | # (1/2TGO - D21)*2(-13) = E * 2(-13) |      |
| 35 |         |       | TGO      | #                                    | (8)  |
| 36 |         | PDDL  | DMP      | # IN PDL(6)                          |      |
| 37 |         |       | TGO      |                                      |      |
| 38 |         |       | RDOT     | # RDOT TGO * 2(-24)                  |      |
| 39 |         | DAD   | DSU      | # R + RDOT TGO                       |      |
| 40 |         |       | /R/MAG   | # R + RDOT TGO - RCO                 |      |
| 41 |         |       | RCO      | # MPAC = -DR *2(-24).                |      |
| 42 |         | PDDL  | DMP      | # -DR IN PDL(8)                      | (10) |
| 43 |         |       | DRDOT    |                                      |      |
| 44 |         |       | 04D      | # D21 DRDOT*2(-24)                   |      |
| 45 |         | DAD   | SL2      | # (D21 DRDOT-DR)*2(-22)              | (8)  |
| 46 |         | DDV   | DDV      |                                      |      |
| 47 |         |       | 06D      | # (D21 DRDOT-DR)/E*2(-9)             |      |
| 48 |         |       | TGO      |                                      |      |
| 49 |         | STORE | PRATE    | # B * 2(8)                           |      |
| 50 |         | BMN   | DLOAD    | # B>0 NOT PERMITTED                  |      |
| 51 |         |       | CHKBmag  |                                      |      |
| 52 |         |       |          |                                      |      |
| 53 |         |       |          |                                      |      |
| 54 |         |       |          |                                      |      |
| 55 |         |       |          |                                      |      |
| 56 |         |       |          |                                      |      |
| 57 |         |       |          |                                      |      |
| 58 |         |       |          |                                      |      |
| 59 |         |       |          |                                      |      |
| 60 |         |       |          |                                      |      |

|    |         |        |          |                                      |
|----|---------|--------|----------|--------------------------------------|
| 1  |         |        | HI6ZEROS |                                      |
| 2  |         | STCALL | PRATE    |                                      |
| 3  |         |        | PROK     |                                      |
| 4  | CHKBAG  | SR4    | DDV      | # B*2(4)                             |
| 5  |         |        | TBUP     | # (B / TAU) * 2(21)                  |
| 6  |         | DSU    | BPL      |                                      |
| 7  |         |        | PRLIMIT  | # ( B / TAU ) = 2(21) MAX.           |
| 8  |         |        | PROK     |                                      |
| 9  |         | DLOAD  | DMP      |                                      |
| 10 |         |        | PRLIMIT  |                                      |
| 11 |         |        | TBUP     | # B MAX. * 2(4)                      |
| 12 |         | SL4    |          | # BMAX*2(8)                          |
| 13 |         | STORE  | PRATE    |                                      |
| 14 | PROK    | DLOAD  |          |                                      |
| 15 |         | DMP    | TGO      |                                      |
| 16 |         |        | DAD      | # YDOT TGO                           |
| 17 |         |        | YDOT     |                                      |
| 18 |         | DSU    | Y        | # Y + YDOT TGO                       |
| 19 |         |        | PDDL     | # Y + YDOT TGO - YCO                 |
| 20 |         |        | YCO      | # MPAC = - DY*(-24.) IN PDL(8) (10)  |
| 21 |         | DMP    | DYDOT    |                                      |
| 22 |         |        | DAD      | # D21 DYDOT - DY (8)                 |
| 23 |         |        | 04D      |                                      |
| 24 |         | SL2    | DDV      | # (D21 DYDOT - DY)/E*2(-9)           |
| 25 |         | DDV    | SETPD    | # (D21 DYDOT - DY)/E TGO*2(8)        |
| 26 |         |        | TGO      | # = D*2(8)                           |
| 27 |         |        | 04       |                                      |
| 28 |         | STORE  | YRATE    |                                      |
| 29 | CONST   | DLOAD  | DMP      | # LOAD B*2(8)                        |
| 30 |         |        | PRATE    | # B D12*2(-9)                        |
| 31 |         | PDDL   | 02D      |                                      |
| 32 |         |        | DDV      | # D12 B IN PDL(4) (6)                |
| 33 |         |        | DRDOT    | # LOAD DRDOT*2(-7)                   |
| 34 |         |        | 00D      | # -DRDOT/L*2(-7)                     |
| 35 |         | SR2    | DSU      | # (-DRDOT/L-D12 B)=A*2(-9) (4)       |
| 36 |         | STADR  |          |                                      |
| 37 |         | STODL  | PCONS    |                                      |
| 38 |         |        | YRATE    | # D*2(8)                             |
| 39 |         | DMP    | PDDL     | # D12 D,EXCH WITH -L IN PDL(0) (2,2) |
| 40 |         | BDDV   | SR2      | # -DYDOT/L*2(-9)                     |
| 41 |         |        | DYDOT    |                                      |
| 42 |         | DSU    |          | # (-DYDOT/L-D12 D)=C*2(-9)           |
| 43 |         |        | 00D      |                                      |
| 44 |         | STORE  | YCONS    |                                      |
| 45 | CMONENT | SETPD  | DLOAD    |                                      |
| 46 |         |        | 00D      |                                      |
| 47 |         |        | 100CS    |                                      |
| 48 |         | DMP    |          |                                      |
| 49 |         | DAD    | PRATE    | # B(T-T0)*2(-9)                      |
| 50 |         |        | DDV      | # (A+B(T-T0))*2(-9)                  |
| 51 |         |        |          |                                      |
| 52 |         |        |          |                                      |
| 53 |         |        |          |                                      |
| 54 |         |        |          |                                      |
| 55 |         |        |          |                                      |
| 56 |         |        |          |                                      |
| 57 |         |        |          |                                      |
| 58 |         |        |          |                                      |
| 59 |         |        |          |                                      |
| 60 |         |        |          |                                      |

|        |       |          |                                         |
|--------|-------|----------|-----------------------------------------|
|        |       | PCONS    | # (A+B(T-T0))/TBUP*2(8)                 |
|        |       | TBUP     |                                         |
|        | SL1   | DSU      |                                         |
|        |       | GEFF     | # ATR*2(9)                              |
|        | STODL | ATR      |                                         |
|        |       | 100CS    |                                         |
|        | DMP   | DAD      |                                         |
|        |       | YRATE    |                                         |
|        |       | YCONS    | # (C+D(T-T0))*2(-9)                     |
|        | DDV   | SL1      |                                         |
|        |       | TBUP     |                                         |
|        | STORE | ATY      | # ATY*2(9)                              |
|        | VXSC  | PDDL     | # ATY UY*2(8) (6)                       |
|        |       | LAXIS    |                                         |
|        |       | ATR      |                                         |
|        | VXSC  | VAD      |                                         |
|        |       | UNIT/R/  |                                         |
|        | VSL1  | PUSH     | # AH*2(9) IN PDL(0) (6)                 |
|        | ABVAL | PDDL     | # AH(2) IN PDL(34)                      |
|        |       | AT       | # AHMAG IN PDL(6) (8)                   |
|        | DSQ   | DSU      | # (AT(2)-AH(2))*2(18)                   |
|        |       | 34D      | # =ATP2*2(18)                           |
|        | PDDL  | PUSH     | # (12)                                  |
|        |       | AT       |                                         |
|        | DSQ   | DSU      | # (AT(2)KR(2)-AH(2))*2(18) (10)         |
|        |       | 34D      | # =ATP3*2(18)                           |
|        | BMN   | DLOAD    | # IF ATP3 NEG,GO TO NO-ATP              |
|        |       | NO-ATP   | # LOAD ATP2, IF ATP3 POS                |
|        |       | 8D       |                                         |
|        | SQRT  | GOTO     | # ATP*2(9)                              |
|        |       | AIMER    |                                         |
| NO-ATP | DLOAD | BDDV     | # KR AT/AH = KH (8)                     |
|        |       | 6D       |                                         |
|        | VXSC  |          | # KH AG*2(9)                            |
|        |       | 00D      |                                         |
|        | STODL | 00D      | # STORE NEW AH IN PDL(0)                |
|        |       | HI6ZEROS |                                         |
| AIMER  | SIGN  |          |                                         |
|        |       | DZDOT    |                                         |
|        | STORE | ATP      |                                         |
|        | VXSC  |          |                                         |
|        |       | ZAXIS1   | # ATP ZAXIS *2(8).                      |
|        | VSL1  | VAD      | # AT*2(0)                               |
|        |       | 00D      |                                         |
|        | STORE | UNFC/2   | # WILL BE OVERWRITTEN IF IN VERT. RISE. |
|        | SETPD | BON      |                                         |
|        |       | 00D      |                                         |
|        |       | FLPI     |                                         |
|        | BON   | P12RET   |                                         |



|    |          |        |             |                                          |
|----|----------|--------|-------------|------------------------------------------|
| 1  |          |        | FLVR        |                                          |
| 2  |          |        | CHECKALT    |                                          |
| 3  |          |        |             |                                          |
| 4  | MAINLINE | VLOAD  | VCOMP       |                                          |
| 5  |          |        | UNIT/R/     |                                          |
| 6  |          | STODL  | UNWC/2      |                                          |
| 7  |          |        | TXO         |                                          |
| 8  |          | DSU    | BPL         |                                          |
| 9  |          |        | PIPTIME     |                                          |
| 10 |          |        | ASCTERM     |                                          |
| 11 |          | BON    |             |                                          |
| 12 |          |        | ROTFLAG     |                                          |
| 13 | CLRFLAG  | CLEAR  | ANG1CHEK    |                                          |
| 14 |          |        | CLEAR       |                                          |
| 15 |          |        | NOR29FLG    | # START R29 IN ASCENT PHASE.             |
| 16 |          |        | XOVINFLG    | # ALLOW X-AXIS OVERRIDE                  |
| 17 | ASCTERM  | EXIT   |             |                                          |
| 18 |          | CA     | FLAGWRD9    |                                          |
| 19 |          | MASK   | FLRCSBIT    |                                          |
| 20 |          | CCS    | A           |                                          |
| 21 |          | TCF    | ASCTERM3    |                                          |
| 22 |          | TC     | INTPRET     |                                          |
| 23 |          | CALL   |             |                                          |
| 24 |          |        | FINDCDUW -2 |                                          |
| 25 | ASCTERM1 | EXIT   |             |                                          |
| 26 | +1       | CA     | FLAGWRD9    | # INSURE THAT THE NOUN 63 DISPLAY IS     |
| 27 |          | MASK   | FLRCSBIT    | # BYPASSED IF WE ARE IN THE RCS TRIMMING |
| 28 |          | CCS    | A           | # MODE OF OPERATION                      |
| 29 |          | TCF    | ASCTERM3    |                                          |
| 30 |          | CA     | FLAGWRD8    | # BYPASS DISPLAYS IF ENGINE FAILURE IS   |
| 31 |          | MASK   | FLUNDBIT    | # INDICATED.                             |
| 32 |          | CCS    | A           |                                          |
| 33 |          | TCF    | ASCTERM3    |                                          |
| 34 |          | CAF    | V06N63*     |                                          |
| 35 |          | TC     | BANKCALL    |                                          |
| 36 |          | CADR   | GODSPR      |                                          |
| 37 |          | TCF    | ASCTERM3    |                                          |
| 38 | ASCTERM2 | EXIT   |             |                                          |
| 39 | ASCTERM3 | TCF    | ENDOFJOB    |                                          |
| 40 | ASCTERM4 | EXIT   |             |                                          |
| 41 |          | INHINT |             |                                          |
| 42 |          | TC     | IBNKCALL    | # NO GUIDANCE THIS CYCLE -- HENCE ZERO   |
| 43 |          | CADR   | ZATTEROR    | # THE DAP COMMANDED ERRORS.              |
| 44 |          | TCF    | ASCTERM1 +1 |                                          |
| 45 |          |        |             |                                          |
| 46 | CHECKALT | DLOAD  | DSU         |                                          |
| 47 |          |        | /R/MAG      |                                          |
| 48 |          |        | /LAND/      |                                          |
| 49 |          | DSU    | BMN         | # IF H LT 25K CHECK Z AXIS ORIENTATION   |
| 50 |          |        | 25KFT       |                                          |
| 51 |          |        | CHECKYAW    |                                          |
| 52 |          |        |             |                                          |
| 53 |          |        |             |                                          |
| 54 |          |        |             |                                          |
| 55 |          |        |             |                                          |
| 56 |          |        |             |                                          |
| 57 |          |        |             |                                          |
| 58 |          |        |             |                                          |
| 59 |          |        |             |                                          |
| 60 |          |        |             |                                          |

|    |          |        |            |    |
|----|----------|--------|------------|----|
| 1  |          |        |            | 1  |
| 2  | EXITVR   | CLEAR  | BON        | 2  |
| 3  |          |        | FLVR       | 3  |
| 4  |          |        | ROTFLAG    | 4  |
| 5  |          |        | MAINLINE   | 5  |
| 6  |          | DLOAD  | DAD        | 6  |
| 7  |          |        | PIPTIME    | 7  |
| 8  |          |        | 10SECS     | 8  |
| 9  |          | STCALL | TXO        | 9  |
| 10 |          |        | MAINLINE   | 10 |
| 11 | EXITVR1  | CLRG0  |            | 11 |
| 12 |          |        | ROTFLAG    | 12 |
| 13 |          |        | EXITVR     | 13 |
| 14 |          |        |            | 14 |
| 15 |          | SETLOC | ASENT1     | 15 |
| 16 |          | BANK   |            | 16 |
| 17 |          | COUNT* | \$\$/ASENT | 17 |
| 18 |          |        |            | 18 |
| 19 | ANGICHEK | VLOAD  | DOT        | 19 |
| 20 |          |        | UNFC/2     | 20 |
| 21 |          |        | XNBPIP     | 21 |
| 22 |          | DSU    | BPL        | 22 |
| 23 |          |        | COSTHET1   | 23 |
| 24 |          |        | OFFROT     | 24 |
| 25 |          | VLOAD  | DOT        | 25 |
| 26 |          |        | XNBPIP     | 26 |
| 27 |          |        | UNIT/R/    | 27 |
| 28 |          | DSU    | BMN        | 28 |
| 29 |          |        | COSTHET2   | 29 |
| 30 |          |        | KEEPVR1    | 30 |
| 31 | OFFROT   | CLRG0  |            | 31 |
| 32 |          |        | ROTFLAG    | 32 |
| 33 |          |        | CLRFLAG    | 33 |
| 34 |          |        |            | 34 |
| 35 |          | BANK   | 7          | 35 |
| 36 |          | SETLOC | ASENT2     | 36 |
| 37 |          | BANK   |            | 37 |
| 38 |          | COUNT* | \$\$/ASENT | 38 |
| 39 |          |        |            | 39 |
| 40 | SETXFLAG | =      | CHECKYAW   | 40 |
| 41 |          |        |            | 41 |
| 42 | CHECKYAW | SET    |            | 42 |
| 43 |          |        |            | 43 |
| 44 |          | DLOAD  | XOVINFLG   | 44 |
| 45 |          |        | VXSC       | 45 |
| 46 |          |        | ATY        | 46 |
| 47 |          | PDDL   | LAXIS      | 47 |
| 48 |          |        | VXSC       | 48 |
| 49 |          |        | ATP        | 49 |
| 50 |          | VAD    | ZAXIS1     | 50 |
| 51 |          | PUSH   | UNIT       | 51 |
| 52 |          |        | DOT        | 52 |
| 53 |          |        |            | 53 |
| 54 |          |        |            | 54 |
| 55 |          |        |            | 55 |
| 56 |          |        |            | 56 |
| 57 |          |        |            | 57 |
| 58 |          |        |            | 58 |
| 59 |          |        |            | 59 |
| 60 |          |        |            | 60 |

# PROHIBIT X-AXIS OVERRIDE

|    |         |        |             |                                          |
|----|---------|--------|-------------|------------------------------------------|
| 1  |         |        | YNBPIP      |                                          |
| 2  |         |        | DSU         |                                          |
| 3  |         | ABS    |             |                                          |
| 4  |         |        | SIN5DEG     |                                          |
| 5  |         | BPL    | DLOAD       |                                          |
| 6  |         |        | KEEPVR      |                                          |
| 7  |         |        | RDOT        |                                          |
| 8  |         | DSU    | BPL         |                                          |
| 9  |         |        | 40FPS       |                                          |
| 10 |         |        | EXITVR1     |                                          |
| 11 |         | GOTO   |             |                                          |
| 12 |         |        | KEEPVR      |                                          |
| 13 |         |        |             |                                          |
| 14 |         | BANK   | 5           |                                          |
| 15 |         | SETLOC | ASENT3      |                                          |
| 16 |         | BANK   |             |                                          |
| 17 |         | COUNT* | \$\$/ASENT  |                                          |
| 18 |         |        |             |                                          |
| 19 | SIN5DEG | 2DEC   | 0.08716 B-2 |                                          |
| 20 | 40FPS   | 2DEC   | 0.12192 B-7 |                                          |
| 21 |         |        |             |                                          |
| 22 |         | BANK   | 14          |                                          |
| 23 |         | SETLOC | ASENT4      |                                          |
| 24 |         | BANK   |             |                                          |
| 25 |         | COUNT* | \$\$/ASENT  |                                          |
| 26 |         |        |             |                                          |
| 27 | KEEPVR  | VLOAD  | STADR       | # RECALL LOSVEC FROM PUSHLIST            |
| 28 |         | STORE  | UNWC/2      |                                          |
| 29 | KEEPVR1 | VLOAD  |             |                                          |
| 30 |         |        | UNIT/R/     |                                          |
| 31 |         | STCALL | UNFC/2      |                                          |
| 32 |         |        | ASCTERM     |                                          |
| 33 |         |        |             |                                          |
| 34 | ENGOFF  | RTB    |             |                                          |
| 35 |         |        | LOADTIME    |                                          |
| 36 |         | DSU    | DAD         |                                          |
| 37 |         |        | PIPTIME     |                                          |
| 38 |         |        | TTOGO       |                                          |
| 39 |         | DCOMP  | EXIT        |                                          |
| 40 |         | TC     | TPAGREE     | # FORCE SIGN AGREEMENT ON MPAC, MPAC +1. |
| 41 |         | CAF    | EBANK7      |                                          |
| 42 |         | TS     | EBANK       |                                          |
| 43 |         | EBANK= | TGO         |                                          |
| 44 |         | INHINT |             |                                          |
| 45 |         | CCS    | MPAC +1     |                                          |
| 46 |         | TCF    | +3          | # C(A) = DT - 1 BIT                      |
| 47 |         | TCF    | +2          | # C(A) = 0                               |
| 48 |         | CAF    | ZERO        | # C(A) = 0                               |
| 49 |         | AD     | BIT1        | # C(A) = 1 BIT OR DT.                    |
| 50 |         |        |             |                                          |
| 51 |         |        |             |                                          |
| 52 |         |        |             |                                          |
| 53 |         |        |             |                                          |
| 54 |         |        |             |                                          |
| 55 |         |        |             |                                          |
| 56 |         |        |             |                                          |
| 57 |         |        |             |                                          |
| 58 |         |        |             |                                          |
| 59 |         |        |             |                                          |
| 60 |         |        |             |                                          |

|    |         |         |          |                                            |    |
|----|---------|---------|----------|--------------------------------------------|----|
| 1  |         |         |          |                                            | 1  |
| 2  |         | TS      | ENGOFFDT |                                            | 2  |
| 3  |         | TC      | TWIDDLE  |                                            | 3  |
| 4  |         | ADRES   | ENGOFF1  |                                            | 4  |
| 5  |         | TC      | PHASCHNG |                                            | 5  |
| 6  |         | OCT     | 47014    |                                            | 6  |
| 7  |         | -GENADR | ENGOFFDT |                                            | 7  |
| 8  |         | EBANK=  | TGO      |                                            | 8  |
| 9  |         | 2CADR   | ENGOFF1  |                                            | 9  |
| 10 |         |         |          |                                            | 10 |
| 11 |         | TC      | INTPRET  |                                            | 11 |
| 12 |         | SET     | GOTO     |                                            | 12 |
| 13 |         |         | IDLEFLAG | # DISABLE DELTA-V MONITOR                  | 13 |
| 14 |         |         | T2TEST   |                                            | 14 |
| 15 |         |         |          |                                            | 15 |
| 16 | ENGOFF1 | TC      | IBNKCALL | # SHUT OFF THE ENGINE.                     | 16 |
| 17 |         | CADR    | ENGINOF2 |                                            | 17 |
| 18 |         |         |          |                                            | 18 |
| 19 |         | CAF     | PRI017   | # SET UP A JOB FOR THE ASCENT GUIDANCE     | 19 |
| 20 |         | TC      | FINDVAC  | # POSTBURN LOGIC.                          | 20 |
| 21 |         | EBANK=  | WHICH    |                                            | 21 |
| 22 |         | 2CADR   | CUTOFF   |                                            | 22 |
| 23 |         |         |          |                                            | 23 |
| 24 |         | TC      | PHASCHNG |                                            | 24 |
| 25 |         | OCT     | 07024    |                                            | 25 |
| 26 |         | OCT     | 17000    |                                            | 26 |
| 27 |         | EBANK=  | TGO      |                                            | 27 |
| 28 |         | 2CADR   | CUTOFF   |                                            | 28 |
| 29 |         |         |          |                                            | 29 |
| 30 |         | TCF     | TASKOVER |                                            | 30 |
| 31 |         |         |          |                                            | 31 |
| 32 | CUTOFF  | TC      | UPFLAG   | # SET FLRCS FLAG.                          | 32 |
| 33 |         | ADRES   | FLRCS    |                                            | 33 |
| 34 |         |         |          |                                            | 34 |
| 35 | -5      | CAF     | V16N63   |                                            | 35 |
| 36 |         | TC      | BANKCALL |                                            | 36 |
| 37 |         | CADR    | GOFLASH  |                                            | 37 |
| 38 |         | TCF     | +3       |                                            | 38 |
| 39 |         | TCF     | CUTOFF1  |                                            | 39 |
| 40 |         | TCF     | -5       |                                            | 40 |
| 41 |         |         |          |                                            | 41 |
| 42 | +3      | TC      | POSTJUMP |                                            | 42 |
| 43 |         | CADR    | TERMASC  |                                            | 43 |
| 44 |         |         |          |                                            | 44 |
| 45 | CUTOFF1 | INHINT  |          |                                            | 45 |
| 46 |         | TC      | IBNKCALL | # ZERO ATTITUDE ERRORS BEFORE REDUCINT DB. | 46 |
| 47 |         | CADR    | ZATTEROR |                                            | 47 |
| 48 |         | TC      | IBNKCALL |                                            | 48 |
| 49 |         | CADR    | SETMINDB |                                            | 49 |
| 50 |         | TC      | POSTJUMP |                                            | 50 |
| 51 |         | CADR    | CUTOFF2  |                                            | 51 |
| 52 |         |         |          |                                            | 52 |
| 53 |         |         |          |                                            | 53 |
| 54 |         |         |          |                                            | 54 |
| 55 |         |         |          |                                            | 55 |
| 56 |         |         |          |                                            | 56 |
| 57 |         |         |          |                                            | 57 |
| 58 |         |         |          |                                            | 58 |
| 59 |         |         |          |                                            | 59 |
| 60 |         |         |          |                                            | 60 |

|    |         |         |                                          |    |
|----|---------|---------|------------------------------------------|----|
| 1  |         |         |                                          | 1  |
| 2  |         |         |                                          | 2  |
| 3  | V16N63  | VN      | 1663                                     | 3  |
| 4  |         | BANK    | 30                                       | 4  |
| 5  |         | SETLOC  | ASENT5                                   | 5  |
| 6  |         | BANK    |                                          | 6  |
| 7  |         | COUNT*  | \$\$/ASENT                               | 7  |
| 8  |         |         |                                          | 8  |
| 9  | CUTOFF2 | TC      | PHASCHNG                                 | 9  |
| 10 |         | OCT     | 04024                                    | 10 |
| 11 |         |         |                                          | 11 |
| 12 |         | CAF     | V16N85C                                  | 12 |
| 13 |         | TC      | BANKCALL                                 | 13 |
| 14 |         | CADR    | GOFLASH                                  | 14 |
| 15 |         | TCF     | TERMASC                                  | 15 |
| 16 |         | TCF     | +2                                       | 16 |
| 17 |         | TCF     | CUTOFF2                                  | 17 |
| 18 |         |         |                                          | 18 |
| 19 | TERMASC | TC      | PHASCHNG                                 | 19 |
| 20 |         | OCT     | 04024                                    | 20 |
| 21 |         |         |                                          | 21 |
| 22 |         | INHINT  | # RESTORE DEADBAND DESIRED BY ASTRONAUT. | 22 |
| 23 |         | TC      | IBNKCALL                                 | 23 |
| 24 |         | CADR    | RESTORDB                                 | 24 |
| 25 |         | TC      | DOWNFLAG                                 | 25 |
| 26 |         | ADRES   | LETABORT                                 | 26 |
| 27 |         | TCF     | GOTOP00H                                 | 27 |
| 28 |         |         |                                          | 28 |
| 29 | V16N85C | VN      | 1685                                     | 29 |
| 30 |         |         |                                          | 30 |
| 31 |         | BANK 27 |                                          | 31 |
| 32 |         | SETLOC  | ASENT1                                   | 32 |
| 33 |         | BANK    |                                          | 33 |
| 34 |         | COUNT*  | \$\$/ASENT                               | 34 |
| 35 |         |         |                                          | 35 |
| 36 | YCOMP   | VLOAD   | DOT                                      | 36 |
| 37 |         |         | UNIT/R/                                  | 37 |
| 38 |         |         | QAXIS                                    | 38 |
| 39 |         | SL2     | DMP                                      | 39 |
| 40 |         |         | RCO                                      | 40 |
| 41 |         | STORE   | Y                                        | 41 |
| 42 |         | RVQ     |                                          | 42 |
| 43 |         |         |                                          | 43 |
| 44 |         | BANK    | 30                                       | 44 |
| 45 |         | SETLOC  | ASENT                                    | 45 |
| 46 |         | BANK    |                                          | 46 |
| 47 |         |         |                                          | 47 |
| 48 |         |         |                                          | 48 |
| 49 |         |         |                                          | 49 |
| 50 |         |         |                                          | 50 |
| 51 |         |         |                                          | 51 |
| 52 |         |         |                                          | 52 |
| 53 |         |         |                                          | 53 |
| 54 |         |         |                                          | 54 |
| 55 |         |         |                                          | 55 |
| 56 |         |         |                                          | 56 |
| 57 |         |         |                                          | 57 |
| 58 |         |         |                                          | 58 |
| 59 |         |         |                                          | 59 |
| 60 |         |         |                                          | 60 |

|    |                   |        |            |                                  |    |
|----|-------------------|--------|------------|----------------------------------|----|
| 1  | # ASCENT GUIDANCE |        |            |                                  | 1  |
| 2  | 100CS             | EQUALS | 2SEC(18)   |                                  | 2  |
| 3  | T2A               | EQUALS | 2SEC(17)   |                                  | 3  |
| 4  | 4SEC(17)          | 2DEC   | 400 B-17   |                                  | 4  |
| 5  | 2SEC(17)          | 2DEC   | 200 B-17   |                                  | 5  |
| 6  | T3                | 2DEC   | 1000 B-17  |                                  | 6  |
| 7  | 6SEC(18)          | 2DEC   | 600 B-18   |                                  | 7  |
| 8  | BIT4H             | OCT    | 10         |                                  | 8  |
| 9  | 2SEC(9)           | 2DEC   | 200 B-9    |                                  | 9  |
| 10 | V06N63*           | VN     | 0663       |                                  | 10 |
| 11 | V06N76            | VN     | 0676       |                                  | 11 |
| 12 | V06N33A           | VN     | 0633       |                                  | 12 |
| 13 |                   |        |            |                                  | 13 |
| 14 |                   | BANK   | 33         |                                  | 14 |
| 15 |                   | SETLOC | ASENT6     |                                  | 15 |
| 16 |                   | BANK   |            |                                  | 16 |
| 17 |                   | COUNT* | \$\$/ASENT |                                  | 17 |
| 18 |                   |        |            |                                  | 18 |
| 19 | KT1               | 2DEC   | 0.5000     |                                  | 19 |
| 20 | PRLIMIT           | 2DEC   | -.0639     | # (B/TBUP)MIN=-.1FT.SEC(-3)      | 20 |
| 21 | MINABDV           | 2DEC   | .0356 B-5  | # 10 PERCENT BIGGER THAN GRAVITY | 21 |
| 22 | 1/DVO             | =      | MASS1      |                                  | 22 |
| 23 |                   |        |            |                                  | 23 |
| 24 |                   |        |            |                                  | 24 |
| 25 |                   |        |            |                                  | 25 |
| 26 |                   |        |            |                                  | 26 |
| 27 |                   |        |            |                                  | 27 |
| 28 |                   |        |            |                                  | 28 |
| 29 |                   |        |            |                                  | 29 |
| 30 |                   |        |            |                                  | 30 |
| 31 |                   |        |            |                                  | 31 |
| 32 |                   |        |            |                                  | 32 |
| 33 |                   |        |            |                                  | 33 |
| 34 |                   |        |            |                                  | 34 |
| 35 |                   |        |            |                                  | 35 |
| 36 |                   |        |            |                                  | 36 |
| 37 |                   |        |            |                                  | 37 |
| 38 |                   |        |            |                                  | 38 |
| 39 |                   |        |            |                                  | 39 |
| 40 |                   |        |            |                                  | 40 |
| 41 |                   |        |            |                                  | 41 |
| 42 |                   |        |            |                                  | 42 |
| 43 |                   |        |            |                                  | 43 |
| 44 |                   |        |            |                                  | 44 |
| 45 |                   |        |            |                                  | 45 |
| 46 |                   |        |            |                                  | 46 |
| 47 |                   |        |            |                                  | 47 |
| 48 |                   |        |            |                                  | 48 |
| 49 |                   |        |            |                                  | 49 |
| 50 |                   |        |            |                                  | 50 |
| 51 |                   |        |            |                                  | 51 |
| 52 |                   |        |            |                                  | 52 |
| 53 |                   |        |            |                                  | 53 |
| 54 |                   |        |            |                                  | 54 |
| 55 |                   |        |            |                                  | 55 |
| 56 |                   |        |            |                                  | 56 |
| 57 |                   |        |            |                                  | 57 |
| 58 |                   |        |            |                                  | 58 |
| 59 |                   |        |            |                                  | 59 |
| 60 |                   |        |            |                                  | 60 |



|    |                                |        |              |    |
|----|--------------------------------|--------|--------------|----|
| 1  | # THE LOGARITHM SUBROUTINE     |        |              | 1  |
| 2  |                                |        |              | 2  |
| 3  |                                |        |              | 3  |
| 4  |                                | BANK   | 24           | 4  |
| 5  |                                | SETLOC | FLOGSUB      | 5  |
| 6  |                                | BANK   |              | 6  |
| 7  |                                |        |              | 7  |
| 8  | # INPUT ..... X IN MPAC        |        |              | 8  |
| 9  | # OUTPUT ..... -LOG(X) IN MPAC |        |              | 9  |
| 10 |                                |        |              | 10 |
| 11 | LOGSUB                         | NORM   | BDSU         | 11 |
| 12 |                                |        | MPAC +6      | 12 |
| 13 |                                |        | NEARONE      | 13 |
| 14 |                                | EXIT   |              | 14 |
| 15 |                                | TC     | POLY         | 15 |
| 16 |                                | DEC    | 6            | 16 |
| 17 |                                | 2DEC   | .0000000060  | 17 |
| 18 |                                | 2DEC   | -.0312514377 | 18 |
| 19 |                                | 2DEC   | -.0155686771 | 19 |
| 20 |                                | 2DEC   | -.0112502068 | 20 |
| 21 |                                | 2DEC   | -.0018545108 | 21 |
| 22 |                                | 2DEC   | -.0286607906 | 22 |
| 23 |                                | 2DEC   | .0385598563  | 23 |
| 24 |                                | 2DEC   | -.0419361902 | 24 |
| 25 |                                |        |              | 25 |
| 26 |                                | CAF    | ZERO         | 26 |
| 27 |                                | TS     | MPAC +2      | 27 |
| 28 |                                | EXTEND |              | 28 |
| 29 |                                | DCA    | CLOG2/32     | 29 |
| 30 |                                | DXCH   | MPAC         | 30 |
| 31 |                                | DXCH   | BUF +1       | 31 |
| 32 |                                | CA     | MPAC +6      | 32 |
| 33 |                                | TC     | SHORTMP      | 33 |
| 34 |                                | DXCH   | MPAC +1      | 34 |
| 35 |                                | DXCH   | MPAC         | 35 |
| 36 |                                | DXCH   | BUF +1       | 36 |
| 37 |                                | DAS    | MPAC         | 37 |
| 38 |                                | TC     | INTPRET      | 38 |
| 39 |                                | DCOMP  | RVQ          | 39 |
| 40 |                                |        |              | 40 |
| 41 | CLOG2/32                       | 2DEC   | .0216608494  | 41 |
| 42 |                                |        |              | 42 |
| 43 |                                |        |              | 43 |
| 44 |                                |        |              | 44 |
| 45 |                                |        |              | 45 |
| 46 |                                |        |              | 46 |
| 47 |                                |        |              | 47 |
| 48 |                                |        |              | 48 |
| 49 |                                |        |              | 49 |
| 50 |                                |        |              | 50 |
| 51 |                                |        |              | 51 |
| 52 |                                |        |              | 52 |
| 53 |                                |        |              | 53 |
| 54 |                                |        |              | 54 |
| 55 |                                |        |              | 55 |
| 56 |                                |        |              | 56 |
| 57 |                                |        |              | 57 |
| 58 |                                |        |              | 58 |
| 59 |                                |        |              | 59 |
| 60 |                                |        |              | 60 |

```
1
2 BANK 37
3 SETLOC SERV1
4 BANK
5
6 EBANK= DVCNTR
7
8 # ***** PREREAD *****
9
10 COUNT* $$/SERV
11
12 PREREAD CAF SEVEN # 5.7 SPOT TO SKIP LASTBIAS AFTER
13 TC GNUFAZE5 # RESTART.
14 CAF PRI021
15 TC NOVAC
16
17 EBANK= NBDX
18 2CADR LASTBIAS # DO LAST GYRO COMPENSATION IN FREE FALL
19
20 BIBIBIAS TC PIPASR +3 # CLEAR + READ PIPS LAST TIME IN FRE5+F133
21 # DO NOT DESTROY VALUE OF PIPTIME1
22
23 CS FLAGWRD7
24 MASK SUPER011 # SET V37FLAG AND AVEGFLAG (BITS 5 AND 6
25 ADS FLAGWRD7 # OF FLAGWRD7)
26
27 CS DRFTBIT
28 MASK FLAGWRD2 # RESET DRIFTFLAG
29 TS FLAGWRD2
30
31 CAF FOUR # INITIALIZE DV MONITOR
32 TS PIPAGE
33
34 CAF ENDJBCAD # POINT OUTROUTE TO END-OF-JOB.
35 TS OUTROUTE
36
37 CAF PRI022
38 TC FINDVAC # TO FIRST ENTRY TO AVERAGE G
39 EBANK= DVCNTR
40 2CADR NORMLIZE
41
42 GOREADAX CA TWO # 5.2SPOT FOR REREADAC AND NORMLIZE
43 TC GNUTFAZ5
44 CA 2SECS # WAIT TWO SECONDS FOR READACCS
45 TC VARDELAY
```



# \*\*\*\*\* READACCS \*\*\*\*\*

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| READACCS | CS     | OCT37771 | # THIS PIECE OF CODING ATTEMPTS TO         |
|          | AD     | TIME5    | # SYNCHRONIZE READACCS WITH THE DIGITAL    |
|          | CCS    | A        | # AUTOPILOT SO THAT A PAXIS RUPT WILL      |
|          | CS     | ONE      | # OCCUR APPROXIMATELY 70 MILLISECONDS      |
|          | TCF    | +2       | # FOLLOWING THE READACCS RUPT. THE 70 MS   |
|          | CA     | ONE      | # OFFSET WAS CHOSEN SO THAT THE PAXIS      |
| +2       | ADS    | TIME5    | # RUPT WOULD NOT OCCUR SIMULTANEOUSLY      |
|          |        |          | # WITH ANY OF THE 8 SUBSEQUENT R10,R11     |
|          |        |          | # INTERRUPTS -- THUS MINIMIZING THE POSS-  |
|          |        |          | # IBILITY OF LOSING DOWNRUPTS.             |
|          | TC     | PIPASR   | # READ THE PIPAS.                          |
| PIPSDONE | CA     | FIVE     |                                            |
|          | TC     | GNUFAZE5 |                                            |
| RED05.5  | CAF    | ONE      |                                            |
|          | TS     | PIPAGE   |                                            |
|          | CA     | PRI020   |                                            |
|          | TC     | FINDVAC  |                                            |
|          | EBANK= | DVCNTR   |                                            |
|          | 2CADR  | SERVICER | # SET UP SERVICER JOB                      |
|          | CA     | BIT9     |                                            |
|          | EXTEND |          |                                            |
|          | WOR    | DSALMOUT | # TURN ON TEST CONNECTOR OUTBIT            |
|          | CA     | FLAGWRD7 |                                            |
|          | MASK   | AVEGFBIT |                                            |
|          | EXTEND |          |                                            |
|          | BZF    | AVEGOUT  | # AVEGFLAG DOWN -- SET UP FINAL EXIT       |
|          | CA     | FLAGWRD6 |                                            |
|          | MASK   | MUNFLBIT |                                            |
|          | EXTEND |          |                                            |
|          | BZF    | MAKEACCS | # MUNFLAG CLEAR -- BYPASS LR AND DISP.     |
|          | CCS    | PHASE2   |                                            |
|          | TCF    | MAKEACCS | # PHASE 2 ACTIVATED -- AVOID MULTIPLE R10. |
|          | CAF    | SEVEN    | # SET PIPCTR FOR 4X/SEC RATE.              |
|          | TS     | PIPCTR   |                                            |
|          | CS     | TIME1    | # SET TBASE2 .05 SECONDS IN THE PAST.      |
|          | AD     | FIVE     |                                            |
|          | AD     | NEG1/2   |                                            |
|          | AD     | NEG1/2   |                                            |
|          | XCH    | TBASE2   |                                            |

|    |          |        |             |                                          |
|----|----------|--------|-------------|------------------------------------------|
| 1  |          |        |             |                                          |
| 2  |          | CAF    | DEC17       | # 2.21SPOT FOR R10,R11                   |
| 3  |          | TS     | L           |                                          |
| 4  |          | COM    |             |                                          |
| 5  |          | DXCH   | -PHASE2     |                                          |
| 6  |          |        |             |                                          |
| 7  |          | CAF    | OCT24       | # FIRST R10,R11 IN .200 SECONDS          |
| 8  |          | TC     | WAITLIST    |                                          |
| 9  |          | EBANK= | UNIT/R/     |                                          |
| 10 |          | 2CADR  | R10,R11     |                                          |
| 11 |          |        |             |                                          |
| 12 | MAKEACCS | CA     | FOUR        |                                          |
| 13 |          | TCF    | GOREADAX    | # DO PHASE CHANGE AND RECALL READACCS    |
| 14 |          |        |             |                                          |
| 15 | AVEGOUT  | EXTEND |             |                                          |
| 16 |          | DCA    | AVOUTCAD    | # SET UP FINAL SERVICER EXIT             |
| 17 |          | DXCH   | AVGEXIT     |                                          |
| 18 |          |        |             |                                          |
| 19 |          | CA     | FOUR        | # SET 5.4 SPOT FOR REREADAC AND SERVICER |
| 20 |          | TC     | GNUTFAZ5    | # IF REREADAC IS CALLED, IT WILL EXIT    |
| 21 |          | TC     | TASKOVER    | # END TASK WITHOUT CALLING READACCS      |
| 22 |          |        |             |                                          |
| 23 | GNUTFAZ5 | TS     | L           | # SAVE INPUT IN L                        |
| 24 |          | CS     | TIME1       |                                          |
| 25 |          | TS     | TBASE5      | # SET TBASE5                             |
| 26 |          | TCF    | +2          |                                          |
| 27 |          |        |             |                                          |
| 28 | GNUFAZE5 | TS     | L           | # SAVE INPUT IN L                        |
| 29 |          | CS     | L           | # -PHASE IN A, PHASE IN L.               |
| 30 |          | DXCH   | -PHASE5     | # SET -PHASE5,PHASE5                     |
| 31 |          | TC     | Q           |                                          |
| 32 |          |        |             |                                          |
| 33 |          | EBANK= | DVCNTR      |                                          |
| 34 | AVOUTCAD | 2CADR  | AVGEND      |                                          |
| 35 |          |        |             |                                          |
| 36 | ENDJBCAD | CADR   | SERVEXIT +2 |                                          |
| 37 |          |        |             |                                          |
| 38 | OCT37771 | OCT    | 37771       |                                          |
| 39 |          |        |             |                                          |
| 40 |          | BANK   | 33          |                                          |
| 41 |          | SETLOC | SERVICES    |                                          |
| 42 |          | BANK   |             |                                          |
| 43 |          |        |             |                                          |
| 44 |          | COUNT* | \$\$/SERV   |                                          |
| 45 |          |        |             |                                          |
| 46 |          |        |             |                                          |
| 47 |          |        |             |                                          |
| 48 |          |        |             |                                          |
| 49 |          |        |             |                                          |
| 50 |          |        |             |                                          |
| 51 |          |        |             |                                          |
| 52 |          |        |             |                                          |
| 53 |          |        |             |                                          |
| 54 |          |        |             |                                          |
| 55 |          |        |             |                                          |
| 56 |          |        |             |                                          |
| 57 |          |        |             |                                          |
| 58 |          |        |             |                                          |
| 59 |          |        |             |                                          |
| 60 |          |        |             |                                          |

# \*\*\*\*\* SERVICER \*\*\*\*\*

SERVICER TC PHASCHNG # RESTART REREADAC + SERVICER

OCT 16035

OCT 20000

EBANK=  
2CADR DVCNTR  
GETABVALCAF PRI031  
TS 1/PIPADT# INITIALIZE 1/PIPADT IN CASE RESTART HAS  
# CAUSED LASTBIAS TO BE SKIPPED.TC BANKCALL  
CADR 1/PIPA

# PIPA COMPENSATION CALL

GETABVAL TC  
VLOAD INTPRET  
ABVAL  
DELVEXIT  
CA MPAC  
TS ABDELV

# ABDELV = CM/SEC\*2(-14).

EXTEND  
MP KPIP  
DXCH ABDVCONV

# ABDVCONV = M/CS \* 2(-5).

EXTEND  
DCA MASS  
DXCH MASS1# INITIALIZE MASS1 IN CASE WE SKIP MASSMON  
# ARE WE ON THE SURFACE?MASSMON CS FLAGWRD8  
MASK SURFFBIT  
EXTEND

BZF MOONSPOT

# YES: BYPASS MASS MESS

CA FLGWRD10  
MASK APSFLBIT  
CCS A

# NO: WHICH VEX SHOULD BE USED?

EXTEND  
DCA APSVEX  
TS Q# IF EXTEND IS EXECUTED, APSVEX --> A,  
# OTHERWISE DPSVEX --> AEXTEND  
DCA ABDVCONV  
EXTEND

OCT10002 DV Q

# WHERE APPROPRIATE VEX RESIDES

EXTEND  
MP MASS  
DAS MASS1

MOONSPOT CA KPIP1

# TP MPAC = ABDELV AT 2(14) CM/SEC

TC SHORTMP

# MULTIPLY BY KPIP1 TO GET

# SERVICER

|    |          |        |          |                                         |    |
|----|----------|--------|----------|-----------------------------------------|----|
| 1  |          |        |          |                                         | 1  |
| 2  |          | DXCH   | MPAC     | # ABDELV AT 2(7) M/CS                   | 2  |
| 3  |          | DAS    | DVTOTAL  | # UPDATE DVTOTAL FOR DISPLAY            | 3  |
| 4  |          |        |          |                                         | 4  |
| 5  |          | TC     | TMPTOSPT |                                         | 5  |
| 6  |          |        |          |                                         | 6  |
| 7  |          | TC     | BANKCALL |                                         | 7  |
| 8  |          | CADR   | QUICTRIG |                                         | 8  |
| 9  |          |        |          |                                         | 9  |
| 10 |          | CAF    | XNBPIPAD |                                         | 10 |
| 11 |          | TC     | BANKCALL |                                         | 11 |
| 12 |          | CADR   | FLESHPOT |                                         | 12 |
| 13 |          | TC     | INTPRET  |                                         | 13 |
| 14 | AVERAGEG | BON    | CALL     |                                         | 14 |
| 15 |          |        | MUNFLAG  |                                         | 15 |
| 16 |          |        | RVBOTH   |                                         | 16 |
| 17 |          |        | CALCRVG  |                                         | 17 |
| 18 |          | EXIT   |          |                                         | 18 |
| 19 | GOSERV   | TC     | QUIKFAZ5 |                                         | 19 |
| 20 |          |        |          |                                         | 20 |
| 21 | COPYCYCL | TC     | COPYCYC  |                                         | 21 |
| 22 |          |        |          |                                         | 22 |
| 23 | #        | CA     | ZERO     | # A IS ZERO ON RETURN FROM COPYCYC      | 23 |
| 24 |          | TS     | PIPATMPX |                                         | 24 |
| 25 |          | TS     | PIPATMPY |                                         | 25 |
| 26 |          | TS     | PIPATMPZ |                                         | 26 |
| 27 |          |        |          |                                         | 27 |
| 28 |          | CS     | STEERBIT | # CLEAR STEERSW PRIOR TO DVMON.         | 28 |
| 29 |          | MASK   | FLAGWRD2 |                                         | 29 |
| 30 |          | TS     | FLAGWRD2 |                                         | 30 |
| 31 |          |        |          |                                         | 31 |
| 32 |          | CAF    | IDLEFBIT | # IS THE IDLE FLAG SET?                 | 32 |
| 33 |          | MASK   | FLAGWRD7 |                                         | 33 |
| 34 |          | CCS    | A        |                                         | 34 |
| 35 |          | TCF    | NODVMON1 | # IDLEFLAG = 1, HENCE SET AUXFLAG TO 0. | 35 |
| 36 |          |        |          |                                         | 36 |
| 37 |          | CS     | FLAGWRD6 |                                         | 37 |
| 38 |          | MASK   | AUXFLBIT |                                         | 38 |
| 39 |          | CCS    | A        |                                         | 39 |
| 40 |          | TCF    | NODVMON2 | # AUXFLAG = 0, HENCE SET AUXFLAG TO 1.  | 40 |
| 41 |          |        |          |                                         | 41 |
| 42 | DVMON    | CS     | DVTHRUSH |                                         | 42 |
| 43 |          | AD     | ABDELV   |                                         | 43 |
| 44 |          | EXTEND |          |                                         | 44 |
| 45 |          | BZMF   | LOTHRUST |                                         | 45 |
| 46 |          |        |          |                                         | 46 |
| 47 |          | CS     | FLAGWRD2 | # SET STEERSW.                          | 47 |
| 48 |          | MASK   | STEERBIT |                                         | 48 |
| 49 |          | ADS    | FLAGWRD2 |                                         | 49 |
| 50 |          |        |          |                                         | 50 |
| 51 | DVCNTSET | CAF    | ONE      | # ALLOW TWO PASSES MAXIMUM NOW THAT     | 51 |
| 52 |          |        |          |                                         | 52 |
| 53 |          |        |          |                                         | 53 |
| 54 |          |        |          |                                         | 54 |
| 55 |          |        |          |                                         | 55 |
| 56 |          |        |          |                                         | 56 |
| 57 |          |        |          |                                         | 57 |
| 58 |          |        |          |                                         | 58 |
| 59 |          |        |          |                                         | 59 |
| 60 |          |        |          |                                         | 60 |

1412THE

1

|          |        |          |                                         |
|----------|--------|----------|-----------------------------------------|
|          | TS     | DVCNTR   | # THRUST HAS BEEN DETECTED.             |
|          | CA     | FLGWRD10 | # BRANCH IF APSFLAG IS SET.             |
|          | MASK   | APSFLBIT |                                         |
|          | CCS    | A        |                                         |
|          | TCF    | USEJETS  |                                         |
|          | CA     | BIT9     | # CHECK GIMBAL FAIL BIT                 |
|          | EXTEND |          |                                         |
|          | RAND   | CHAN32   |                                         |
|          | EXTEND |          |                                         |
|          | BZF    | USEJETS  |                                         |
| USEGTS   | CS     | USEQRJTS |                                         |
|          | MASK   | DAPBOOLS |                                         |
|          | TS     | DAPBOOLS |                                         |
|          | TCF    | SERVOUT  |                                         |
| NODVMON1 | CS     | AUXFLBIT | # SET AUXFLAG TO 0.                     |
|          | MASK   | FLAGWRD6 |                                         |
|          | TS     | FLAGWRD6 |                                         |
| NODVMON2 | TCF    | USEJETS  |                                         |
|          | CS     | FLAGWRD6 | # SET AUXFLAG TO 1.                     |
|          | MASK   | AUXFLBIT |                                         |
|          | ADS    | FLAGWRD6 |                                         |
|          | TCF    | USEJETS  |                                         |
| LOTHRUST | TC     | QUIKFAZ5 |                                         |
|          | CCS    | DVCNTR   |                                         |
|          | TCF    | DECCNTR  |                                         |
|          | CCS    | PHASE4   | # COMFAIL JOB ACTIVE?                   |
|          | TCF    | SERVOUT  | # YES: WON'T NEED ANOTHER.              |
|          | TC     | PHASCHNG | # 4.37SPOT FOR COMFAIL.                 |
|          | OCT    | 00374    |                                         |
|          | CAF    | PRI025   |                                         |
|          | TC     | NOVAC    |                                         |
|          | EBANK= | WHICH    |                                         |
|          | 2CADR  | COMFAIL  |                                         |
|          | TCF    | SERVOUT  |                                         |
| DECCNTR  | TS     | DVCNTR1  |                                         |
|          | TC     | QUIKFAZ5 |                                         |
|          | CA     | DVCNTR1  |                                         |
|          | TS     | DVCNTR   |                                         |
|          | INHINT |          |                                         |
|          | TC     | IBNKCALL | # IF THRUST IS LOW, NO STEERING IS DONE |

USEJETS

CADR  
CSSTOPRATE  
DAPBOOLS

# AND THE DESIRED RATES ARE SET TO ZERO.

SERVOUT

MASK  
ADS  
RELINTUSEQRJTS  
DAPBOOLSTC  
CADRBANKCALL  
1/ACCSCA  
MASK  
TSPRIORITY  
LOW9  
PUSHLOCZL  
DXCH

FIXLOC

# FIXLOC AND DVFIND

TC  
EXTEND  
DCAQUIKFAZ5  
AVGEXIT# EXIT TO SELECTED ROUTINE WHETHER THERE  
# IS THRUST OR NOT. THE STATE OF STEERSW  
# WILL CONVEY THIS INFORMATION.

DXCH

Z

XNBPIPAD

ECADR

XNBPIP

BANK  
SETLOC32  
SERV2BANK  
COUNT\*

\$\$/SERV

AVGEND

CA  
TSPIPTIME +1  
1/PIPADT# FINAL AVERAGE G EXIT  
# SET UP FREE FALL GYRO COMPENSATION.TC  
ADRESUPFLAG  
DRIFTFLG

# SET DRIFT FLAG.

TC  
CADRBANKCALL  
PIPFREECS  
EXTEND  
WANDBIT9  
DSALMOUTTC  
OCT  
OCT  
OCT2PHSCHNG  
5  
05022  
20000# GROUP 5 OFF  
# GROUP 2 ONTC  
SETINTPRET  
CLEAR

NOR29FLG

# SHUT OFF R29 WHEN SERVICER ENDS.  
# SHUT OFF R10 WHEN SERVICER ENDS.  
# RESET MUNFLAG.

CLEAR

SWANDISP  
CALL  
MUNFLAG



# SERVICER

1412THE

|    |          |       |          |                            |    |
|----|----------|-------|----------|----------------------------|----|
| 1  |          |       |          |                            | 1  |
| 2  |          |       | AVETOMID |                            | 2  |
| 3  |          | CLEAR | EXIT     |                            | 3  |
| 4  |          |       | V37FLAG  |                            | 4  |
| 5  | AVERTRN  | CA    | OUTROUTE | # RETURN TO DESIRED POINT. | 5  |
| 6  |          | TC    | BANKJUMP |                            | 6  |
| 7  |          |       |          |                            | 7  |
| 8  | OUTGOAVE | =     | AVERTRN  |                            | 8  |
| 9  | DVCNTR1  | =     | MASS1    |                            | 9  |
| 10 |          |       |          |                            | 10 |
| 11 |          |       |          |                            | 11 |
| 12 |          |       |          |                            | 12 |
| 13 |          |       |          |                            | 13 |
| 14 |          |       |          |                            | 14 |
| 15 |          |       |          |                            | 15 |
| 16 |          |       |          |                            | 16 |
| 17 |          |       |          |                            | 17 |
| 18 |          |       |          |                            | 18 |
| 19 |          |       |          |                            | 19 |
| 20 |          |       |          |                            | 20 |
| 21 |          |       |          |                            | 21 |
| 22 |          |       |          |                            | 22 |
| 23 |          |       |          |                            | 23 |
| 24 |          |       |          |                            | 24 |
| 25 |          |       |          |                            | 25 |
| 26 |          |       |          |                            | 26 |
| 27 |          |       |          |                            | 27 |
| 28 |          |       |          |                            | 28 |
| 29 |          |       |          |                            | 29 |
| 30 |          |       |          |                            | 30 |
| 31 |          |       |          |                            | 31 |
| 32 |          |       |          |                            | 32 |
| 33 |          |       |          |                            | 33 |
| 34 |          |       |          |                            | 34 |
| 35 |          |       |          |                            | 35 |
| 36 |          |       |          |                            | 36 |
| 37 |          |       |          |                            | 37 |
| 38 |          |       |          |                            | 38 |
| 39 |          |       |          |                            | 39 |
| 40 |          |       |          |                            | 40 |
| 41 |          |       |          |                            | 41 |
| 42 |          |       |          |                            | 42 |
| 43 |          |       |          |                            | 43 |
| 44 |          |       |          |                            | 44 |
| 45 |          |       |          |                            | 45 |
| 46 |          |       |          |                            | 46 |
| 47 |          |       |          |                            | 47 |
| 48 |          |       |          |                            | 48 |
| 49 |          |       |          |                            | 49 |
| 50 |          |       |          |                            | 50 |
| 51 |          |       |          |                            | 51 |
| 52 |          |       |          |                            | 52 |
| 53 |          |       |          |                            | 53 |
| 54 |          |       |          |                            | 54 |
| 55 |          |       |          |                            | 55 |
| 56 |          |       |          |                            | 56 |
| 57 |          |       |          |                            | 57 |
| 58 |          |       |          |                            | 58 |
| 59 |          |       |          |                            | 59 |
| 60 |          |       |          |                            | 60 |

|    |          |        |           |                                           |    |
|----|----------|--------|-----------|-------------------------------------------|----|
| 1  |          |        |           |                                           | 1  |
| 2  |          | SETLOC | SERV3     |                                           | 2  |
| 3  |          | BANK   |           |                                           | 3  |
| 4  |          | COUNT* | \$\$/SERV |                                           | 4  |
| 5  |          |        |           |                                           | 5  |
| 6  | SERVIDLE | EXTEND |           | # DISCONNECT SERVICER FROM ALL GUIDANCE   | 6  |
| 7  |          | DCA    | SVEXTADR  |                                           | 7  |
| 8  |          | DXCH   | AVGEXIT   |                                           | 8  |
| 9  |          |        |           |                                           | 9  |
| 10 |          | CS     | FLAGWRD7  | # DISCONNECT THE DELTA-V MONITOR          | 10 |
| 11 |          | MASK   | IDLEFBIT  |                                           | 11 |
| 12 |          | ADS    | FLAGWRD7  |                                           | 12 |
| 13 |          |        |           |                                           | 13 |
| 14 |          | CAF    | LRBYBIT   | # TERMINATE R12 IS RUNNING.               | 14 |
| 15 |          | TS     | FLGWRD11  |                                           | 15 |
| 16 |          |        |           |                                           | 16 |
| 17 |          | EXTEND |           |                                           | 17 |
| 18 |          | DCA    | NEGO      |                                           | 18 |
| 19 |          | DXCH   | -PHASE1   |                                           | 19 |
| 20 |          |        |           |                                           | 20 |
| 21 |          | CA     | FLAGWRD6  | # DO NOT TURN OFF PHASE 2 IF MUNFLAG SET. | 21 |
| 22 |          | MASK   | MUNFLBIT  |                                           | 22 |
| 23 |          | CCS    | A         |                                           | 23 |
| 24 |          | TCF    | +4        |                                           | 24 |
| 25 |          |        |           |                                           | 25 |
| 26 |          | EXTEND |           |                                           | 26 |
| 27 |          | DCA    | NEGO      |                                           | 27 |
| 28 |          | DXCH   | -PHASE2   |                                           | 28 |
| 29 |          |        |           |                                           | 29 |
| 30 | +4       | EXTEND |           |                                           | 30 |
| 31 |          | DCA    | NEGO      |                                           | 31 |
| 32 |          | DXCH   | -PHASE3   |                                           | 32 |
| 33 |          |        |           |                                           | 33 |
| 34 |          | EXTEND |           |                                           | 34 |
| 35 |          | DCA    | NEGO      |                                           | 35 |
| 36 |          | DXCH   | -PHASE6   |                                           | 36 |
| 37 |          |        |           |                                           | 37 |
| 38 |          | CAF    | OCT33     | # 4.33SPOT FOR GOP00FIX                   | 38 |
| 39 |          | TS     | L         |                                           | 39 |
| 40 |          | COM    |           |                                           | 40 |
| 41 |          | DXCH   | -PHASE4   |                                           | 41 |
| 42 |          |        |           |                                           | 42 |
| 43 |          | TCF    | WHIMPER   | # PERFORM A SOFTWARE RESTART AND PROCEED  | 43 |
| 44 |          |        |           | # TO GOTOP00H WHILE SERVICER CONTINUES TO | 44 |
| 45 |          |        |           | # RUN, ALBEIT IN A GROUND STATE WHERE     | 45 |
| 46 |          |        |           | # ONLY STATE-VECTOR DEPENDENT FUNCTIONS   | 46 |
| 47 |          |        |           | # ARE MAINTAINED.                         | 47 |
| 48 |          |        |           |                                           | 48 |
| 49 |          | EBANK= | DVCNTR    |                                           | 49 |
| 50 |          |        |           |                                           | 50 |
| 51 |          |        |           |                                           | 51 |
| 52 |          |        |           |                                           | 52 |
| 53 |          |        |           |                                           | 53 |
| 54 |          |        |           |                                           | 54 |
| 55 |          |        |           |                                           | 55 |
| 56 |          |        |           |                                           | 56 |
| 57 |          |        |           |                                           | 57 |
| 58 |          |        |           |                                           | 58 |
| 59 |          |        |           |                                           | 59 |
| 60 |          |        |           |                                           | 60 |





# SERVICER

|    |          |        |           |    |
|----|----------|--------|-----------|----|
| 1  |          |        |           | 1  |
| 2  | SVEXTADR | 2CADR  | SERVEXIT  | 2  |
| 3  |          |        |           | 3  |
| 4  |          | BANK   | 32        | 4  |
| 5  |          | SETLOC | SERV      | 5  |
| 6  |          | BANK   |           | 6  |
| 7  |          | COUNT* | \$\$/SERV | 7  |
| 8  |          |        |           | 8  |
| 9  | SERVEXIT | TC     | PHASCHNG  | 9  |
| 10 |          | OCT    | 00035     | 10 |
| 11 |          |        |           | 11 |
| 12 | +2       | TCF    | ENDOFJOB  | 12 |
| 13 |          |        |           | 13 |
| 14 |          | BANK   | 23        | 14 |
| 15 |          | SETLOC | NORMLIZ   | 15 |
| 16 |          | BANK   |           | 16 |
| 17 |          |        |           | 17 |
| 18 |          | COUNT* | \$\$/SERV | 18 |
| 19 |          |        |           | 19 |
| 20 |          |        |           | 20 |
| 21 |          |        |           | 21 |
| 22 |          |        |           | 22 |
| 23 |          |        |           | 23 |
| 24 |          |        |           | 24 |
| 25 |          |        |           | 25 |
| 26 |          |        |           | 26 |
| 27 |          |        |           | 27 |
| 28 |          |        |           | 28 |
| 29 |          |        |           | 29 |
| 30 |          |        |           | 30 |
| 31 |          |        |           | 31 |
| 32 |          |        |           | 32 |
| 33 |          |        |           | 33 |
| 34 |          |        |           | 34 |
| 35 |          |        |           | 35 |
| 36 |          |        |           | 36 |
| 37 |          |        |           | 37 |
| 38 |          |        |           | 38 |
| 39 |          |        |           | 39 |
| 40 |          |        |           | 40 |
| 41 |          |        |           | 41 |
| 42 |          |        |           | 42 |
| 43 |          |        |           | 43 |
| 44 |          |        |           | 44 |
| 45 |          |        |           | 45 |
| 46 |          |        |           | 46 |
| 47 |          |        |           | 47 |
| 48 |          |        |           | 48 |
| 49 |          |        |           | 49 |
| 50 |          |        |           | 50 |
| 51 |          |        |           | 51 |
| 52 |          |        |           | 52 |
| 53 |          |        |           | 53 |
| 54 |          |        |           | 54 |
| 55 |          |        |           | 55 |
| 56 |          |        |           | 56 |
| 57 |          |        |           | 57 |
| 58 |          |        |           | 58 |
| 59 |          |        |           | 59 |
| 60 |          |        |           | 60 |

# SERVICER

|    |               |        |                                |    |
|----|---------------|--------|--------------------------------|----|
| 1  |               |        |                                | 1  |
| 2  | NORMLIZE      | TC     | INTPRET                        | 2  |
| 3  |               | VLOAD  | BOFF                           | 3  |
| 4  |               |        | RN1                            | 4  |
| 5  |               |        | MUNFLAG                        | 5  |
| 6  |               |        | NORMLIZ1                       | 6  |
| 7  |               | VSL6   | MXV                            | 7  |
| 8  |               |        | REFSMMAT                       | 8  |
| 9  |               | STCALL | R                              | 9  |
| 10 |               |        | MUNGRAV                        | 10 |
| 11 |               | VLOAD  | VSL1                           | 11 |
| 12 |               |        | VN1                            | 12 |
| 13 |               | MXV    |                                | 13 |
| 14 |               |        | REFSMMAT                       | 14 |
| 15 |               | STOVL  | V                              | 15 |
| 16 |               |        | V(CSM)                         | 16 |
| 17 |               | VXV    | UNIT                           | 17 |
| 18 |               |        | R(CSM)                         | 18 |
| 19 | ASCSPOT       | STORE  | UHYP                           | 19 |
| 20 |               | EXIT   |                                | 20 |
| 21 |               | EXTEND | # MAKE SURE GROUP 2 IS OFF     | 21 |
| 22 |               | DCA    | NEGO                           | 22 |
| 23 |               | DXCH   | -PHASE2                        | 23 |
| 24 |               |        |                                | 24 |
| 25 |               | TC     | POSTJUMP                       | 25 |
| 26 |               | CADR   | NORMLIZ2                       | 26 |
| 27 |               |        |                                | 27 |
| 28 |               | BANK   | 33                             | 28 |
| 29 |               | SETLOC | SERVICES                       | 29 |
| 30 |               | BANK   |                                | 30 |
| 31 |               | COUNT* | \$\$/SERV                      | 31 |
| 32 | NORMLIZ1      |        |                                | 32 |
| 33 |               | CALL   |                                | 33 |
| 34 |               |        | CALCGRV                        | 34 |
| 35 |               | EXIT   |                                | 35 |
| 36 |               |        |                                | 36 |
| 37 | NORMLIZ2      | CA     | EIGHTEEN                       | 37 |
| 38 |               | TC     | COPYCYC +1                     | 38 |
| 39 |               | TC     | ENDOFJOB                       | 39 |
| 40 |               |        | # DO NOT COPY MASS IN NORMLIZE | 40 |
| 41 | COPYCYC<br>+1 | CA     | OCT24                          | 41 |
| 42 |               | INHINT | # DEC 20                       | 42 |
| 43 | +2            | MASK   | NEG1                           | 43 |
| 44 |               | TS     | ITEMP1                         | 44 |
| 45 |               | EXTEND | # REDUCE BY 1 IF ODD           | 45 |
| 46 |               | INDEX  | ITEMP1                         | 46 |
| 47 |               | DCA    | RN1                            | 47 |
| 48 |               | INDEX  | ITEMP1                         | 48 |
| 49 |               |        |                                | 49 |
| 50 |               |        |                                | 50 |
| 51 |               |        |                                | 51 |
| 52 |               |        |                                | 52 |
| 53 |               |        |                                | 53 |
| 54 |               |        |                                | 54 |
| 55 |               |        |                                | 55 |
| 56 |               |        |                                | 56 |
| 57 |               |        |                                | 57 |
| 58 |               |        |                                | 58 |
| 59 |               |        |                                | 59 |
| 60 |               |        |                                | 60 |



# SERVICER

PAGE 868



|    |          |      |                       |    |
|----|----------|------|-----------------------|----|
| 1  |          |      |                       | 1  |
| 2  |          | DXCH | RN                    | 2  |
| 3  |          | CCS  | ITEMP1                | 3  |
| 4  |          | TCF  | COPYCYC +2            | 4  |
| 5  |          | TC   | Q                     | 5  |
| 6  |          |      | # RETURN UNDER INHINT | 6  |
| 7  | EIGHTEEN | DEC  | 18                    | 7  |
| 8  |          |      |                       | 8  |
| 9  |          |      |                       | 9  |
| 10 |          |      |                       | 10 |
| 11 |          |      |                       | 11 |
| 12 |          |      |                       | 12 |
| 13 |          |      |                       | 13 |
| 14 |          |      |                       | 14 |
| 15 |          |      |                       | 15 |
| 16 |          |      |                       | 16 |
| 17 |          |      |                       | 17 |
| 18 |          |      |                       | 18 |
| 19 |          |      |                       | 19 |
| 20 |          |      |                       | 20 |
| 21 |          |      |                       | 21 |
| 22 |          |      |                       | 22 |
| 23 |          |      |                       | 23 |
| 24 |          |      |                       | 24 |
| 25 |          |      |                       | 25 |
| 26 |          |      |                       | 26 |
| 27 |          |      |                       | 27 |
| 28 |          |      |                       | 28 |
| 29 |          |      |                       | 29 |
| 30 |          |      |                       | 30 |
| 31 |          |      |                       | 31 |
| 32 |          |      |                       | 32 |
| 33 |          |      |                       | 33 |
| 34 |          |      |                       | 34 |
| 35 |          |      |                       | 35 |
| 36 |          |      |                       | 36 |
| 37 |          |      |                       | 37 |
| 38 |          |      |                       | 38 |
| 39 |          |      |                       | 39 |
| 40 |          |      |                       | 40 |
| 41 |          |      |                       | 41 |
| 42 |          |      |                       | 42 |
| 43 |          |      |                       | 43 |
| 44 |          |      |                       | 44 |
| 45 |          |      |                       | 45 |
| 46 |          |      |                       | 46 |
| 47 |          |      |                       | 47 |
| 48 |          |      |                       | 48 |
| 49 |          |      |                       | 49 |
| 50 |          |      |                       | 50 |
| 51 |          |      |                       | 51 |
| 52 |          |      |                       | 52 |
| 53 |          |      |                       | 53 |
| 54 |          |      |                       | 54 |
| 55 |          |      |                       | 55 |
| 56 |          |      |                       | 56 |
| 57 |          |      |                       | 57 |
| 58 |          |      |                       | 58 |
| 59 |          |      |                       | 59 |
| 60 |          |      |                       | 60 |

1412THE

```
***** PIPA READER *****
MOD NO. 00 BY D. LICKLY, DEC. 9 1966

#
FUNCTIONAL DESCRIPTION
SUBROUTINE TO READ PIPA COUNTERS, TRYING TO BE VERY CAREFUL SO THAT WILL BE RESTARTABLE.
PIPA READINGS ARE STORED IN THE VECTOR DELV. THE HIGH ORDER PART OF EACH COMPONENT CONTAINS THE PIPA READING,
RESTARTS BEGIN AT REREADAC.
#
AT THE END OF THE PIPA READER THE CDUS ARE READ AND STORED AS A
VECTOR IN CDUTEMP. THE HIGH ORDER PART OF EACH COMPONENT CONTAINS
THE CDU READING IN 25 COMP IN THE ORDER CDUX,Y,Z. THE THRUST
VECTOR ESTIMATOR IN FINDCDUD REQUIRES THE CDUS BE READ AT PIPTIME.
#
CALLING SEQUENCE AND EXIT
CALL VIA TC, ISWCALL, ETC.
EXIT IS VIA Q.
#
INPUT
INPUT IS THROUGH THE COUNTERS PIPAX, PIPAY, PIPAZ, AND TIME2.
#
OUTPUT
HIGH ORDER COMPONENTS OF THE VECTOR DELV CONTAIN THE PIPA READINGS.
PIPTIME CONTAINS TIME OF PIPA READING.
#
DEBRIS (ERASABLE LOCATIONS DESTROYED BY PROGRAM)
TEMX, TEMY, TEMZ, PIPAGE

 BANK 37
 SETLOC SERV1
 BANK

 COUNT* $$/SERV

PIPASR EXTEND
```

| 2THE |    |
|------|----|
|      | 1  |
|      | 2  |
|      | 3  |
|      | 4  |
|      | 5  |
|      | 6  |
|      | 7  |
|      | 8  |
|      | 9  |
|      | 10 |
|      | 11 |
|      | 12 |
|      | 13 |
|      | 14 |
|      | 15 |
|      | 16 |
|      | 17 |
|      | 18 |
|      | 19 |
|      | 20 |
|      | 21 |
|      | 22 |
|      | 23 |
|      | 24 |
|      | 25 |
|      | 26 |
|      | 27 |
|      | 28 |
|      | 29 |
|      | 30 |
|      | 31 |
|      | 32 |
|      | 33 |
|      | 34 |
|      | 35 |
|      | 36 |
|      | 37 |
|      | 38 |
|      | 39 |
|      | 40 |
|      | 41 |
|      | 42 |
|      | 43 |
|      | 44 |
|      | 45 |
|      | 46 |
|      | 47 |
|      | 48 |
|      | 49 |
|      | 50 |
|      | 51 |
|      | 52 |
|      | 53 |
|      | 54 |
|      | 55 |
|      | 56 |
|      | 57 |
|      | 58 |
|      | 59 |
|      | 60 |
|      | 61 |
|      | 62 |
|      | 63 |
|      | 64 |
|      | 65 |

# SERVICER

|          |                         |                               |                                                  |
|----------|-------------------------|-------------------------------|--------------------------------------------------|
| REREADAC | CCS<br>TCF              | PIPAGE<br>READACCS            | # PIP READING NOT STARTED. GO TO BEGINNING       |
|          | CAF<br>TS               | DONEADR<br>Q                  | # SET UP RETURN FROM PIPASR                      |
|          | CCS<br>TCF              | DELVZ<br>REPIP4               | # Z DONE, GO DO CDUS                             |
|          | TCF<br>TCF<br>TCF       | +3<br>REPIP4<br>REPIP4        | # Z NOT DONE, CHECK Y.                           |
|          | ZL<br>CCS               | DELVY                         |                                                  |
|          | TCF<br>TCF<br>TCF       | +3<br>CHKTEMX<br>+1           | # Y NOT DONE, CHECK X.                           |
|          | LXCH                    | PIPAZ                         | # Y DONE, ZERO Z PIP.                            |
|          | CCS<br>CS<br>TCF<br>TCF | TEMZ<br>TEMZ<br>DODELVZ<br>-2 | # TEMZ NOT = -0, CONTAINS -PIPAZ VALUE.          |
|          | LXCH<br>TCF             | DELVZ<br>REPIP4               | # TEMZ = -0, L HAS ZPIP VALUE.                   |
| CHKTEMX  | CCS<br>CS<br>TCF        | TEMX<br>TEMX<br>+3            | # HAS THIS CHANGED<br># YES<br># YES             |
|          | TCF<br>TCF<br>TS        | -2<br>REPIP1<br>DELVX         | # YES<br># NO                                    |
|          | CS<br>TS                | TEMY<br>DELVY                 |                                                  |
|          | CS<br>DXCH              | ZERO<br>PIPAX                 | # ZERO X AND Y PIPS<br># L STILL ZERO FROM ABOVE |
|          | TCF                     | REPIP3                        |                                                  |
| DONEADR  | GENADR                  | PIPSDONE                      |                                                  |

```
1
2 BANK 33
3 SETLOC SERVICES
4 BANK
5
6 COUNT* $$/SERV
7
8 TMPTOSPT CA CDUTEMPY # THIS SUBROUTINE, CALLED BY AN RTB FROM
9 TS CDUSPOTY # INTERPRETIVE, LOADS THE CDUS CORRESPON-
10 CA CDUTEMPZ # DING TO PIPTIME INTO THE CDUSPOT VECTOR.
11 TS CDUSPOTZ
12 CA CDUTEMPX
13 TS CDUSPOTX
14 TC Q
```

```
16 # LRHTASK IS A WAITLIST TASK SET BY READACCS DURING THE DESCENT BRAKING
17 # PHASE WHEN THE ALT TO THE LUNAR SURFACE IS LESS THAN 25,000 FT. THIS
18 # TASK CLEARS THE ALTITUDE MEASUREMENT MADE DISCRETE AND INITIATES THE
19 # LANDING RADAR MEASUREMENT JOB (LRHJOB) TO TAKE A ALTITUDE MEASUREMENT
20 # 50 MS PRIOR TO THE NEXT READACCS TASK.
```

```
22 BANK 21
23 SETLOC R10
24 BANK
25
26 COUNT* $$/SERV
27
28 LRHTASK CS FLGWRD11
29 MASK LRBYBIT
30 EXTEND
31 BZF GRP2OFF # LR BYPASS SET -- BYPASS ALL LR READING.
32
33 CA READLBIT
34 MASK FLGWRD11 # IS READLR FLAG SET?
35 EXTEND
36 BZF GRP2OFF # NO. BYPASS LR READ.
37
38 CS FLGWRD11
39 MASK NOLRRBIT # IS LR READ INHIBITED?
40 EXTEND
41 BZF GRP2OFF # YES. BYPASS LR READ.
42
43 CA PRI032 # LR READ OK. SET JOB TO DO IT
44 TC NOVAC # ABOUT 50 MS. PRIOR TO PIPA READ.
45 EBANK= HMEAS
46 2CADR LRHJOB
```

```
48 GRP2OFF EXTEND
49 DCA NEG0
50 DXCH -PHASE2
51 TCF R10,R11A
```

```
53 BANK 33
54 SETLOC SERVICES
55 BANK
```

|    |                                                                           |        |          |                                   |    |
|----|---------------------------------------------------------------------------|--------|----------|-----------------------------------|----|
| 1  |                                                                           |        |          |                                   | 1  |
| 2  | COUNT* \$\$/SERV                                                          |        |          |                                   | 2  |
| 3  |                                                                           |        |          |                                   | 3  |
| 4  | # HIGATASK IS ENTERED APPROXIMATELY 6 SECS PRIOR TO HIGATE DURING THE     |        |          |                                   | 4  |
| 5  | # DESCENT PHASE. HIGATASK SETS THE HIGATE FLAG (BIT11) AND THE LR INHIBIT |        |          |                                   | 5  |
| 6  | # FLAG (BIT10) IN LRSTAT. THE HIGATJOB IS SET UP TO REPOSITION THE LR     |        |          |                                   | 6  |
| 7  | # ANTENNA FROM POSITION 1 TO POSITION 2. IF THE REPOSITIONING IS          |        |          |                                   | 7  |
| 8  | # SUCCESSFUL THE ALT BEAM AND VELOCITY BEAMS ARE TRANSFORMED TO THE NEW   |        |          |                                   | 8  |
| 9  | # ORIENTATION IN NB COORDINATES AND STORED IN ERASABLE.                   |        |          |                                   | 9  |
| 10 |                                                                           |        |          |                                   | 10 |
| 11 | HIGATASK                                                                  | INHINT |          |                                   | 11 |
| 12 |                                                                           | CS     | PRI03    | # SET HIGATE AND LR INHIBIT FLAGS | 12 |
| 13 |                                                                           | MASK   | FLGWRD11 |                                   | 13 |
| 14 |                                                                           | AD     | PRI03    |                                   | 14 |
| 15 |                                                                           | TS     | FLGWRD11 |                                   | 15 |
| 16 |                                                                           | CAF    | PRI032   |                                   | 16 |
| 17 |                                                                           | TC     | FINDVAC  | # SET LR POSITIONING JOB (POS2)   | 17 |
| 18 |                                                                           | EBANK= | HMEAS    |                                   | 18 |
| 19 |                                                                           | 2CADR  | HIGATJOB |                                   | 19 |
| 20 |                                                                           |        |          |                                   | 20 |
| 21 |                                                                           | TCF    | CONTSERV | # CONTINUE SERVICER               | 21 |
| 22 |                                                                           |        |          |                                   | 22 |
| 23 |                                                                           |        |          |                                   | 23 |
| 24 |                                                                           |        |          |                                   | 24 |
| 25 |                                                                           |        |          |                                   | 25 |
| 26 |                                                                           |        |          |                                   | 26 |
| 27 |                                                                           |        |          |                                   | 27 |
| 28 |                                                                           |        |          |                                   | 28 |
| 29 |                                                                           |        |          |                                   | 29 |
| 30 |                                                                           |        |          |                                   | 30 |
| 31 |                                                                           |        |          |                                   | 31 |
| 32 |                                                                           |        |          |                                   | 32 |
| 33 |                                                                           |        |          |                                   | 33 |
| 34 |                                                                           |        |          |                                   | 34 |
| 35 |                                                                           |        |          |                                   | 35 |
| 36 |                                                                           |        |          |                                   | 36 |
| 37 |                                                                           |        |          |                                   | 37 |
| 38 |                                                                           |        |          |                                   | 38 |
| 39 |                                                                           |        |          |                                   | 39 |
| 40 |                                                                           |        |          |                                   | 40 |
| 41 |                                                                           |        |          |                                   | 41 |
| 42 |                                                                           |        |          |                                   | 42 |
| 43 |                                                                           |        |          |                                   | 43 |
| 44 |                                                                           |        |          |                                   | 44 |
| 45 |                                                                           |        |          |                                   | 45 |
| 46 |                                                                           |        |          |                                   | 46 |
| 47 |                                                                           |        |          |                                   | 47 |
| 48 |                                                                           |        |          |                                   | 48 |
| 49 |                                                                           |        |          |                                   | 49 |
| 50 |                                                                           |        |          |                                   | 50 |
| 51 |                                                                           |        |          |                                   | 51 |
| 52 |                                                                           |        |          |                                   | 52 |
| 53 |                                                                           |        |          |                                   | 53 |
| 54 |                                                                           |        |          |                                   | 54 |
| 55 |                                                                           |        |          |                                   | 55 |
| 56 |                                                                           |        |          |                                   | 56 |
| 57 |                                                                           |        |          |                                   | 57 |
| 58 |                                                                           |        |          |                                   | 58 |
| 59 |                                                                           |        |          |                                   | 59 |
| 60 |                                                                           |        |          |                                   | 60 |



# MUNRETRN IS THE RETURN LOC FROM SPECIAL AVE G ROUTINE (MUNRVG)

MUNRETRN EXIT

CS FLGWRD11

MASK LRBYBIT

EXTEND

BZF COPYCYC1

# BYPASS LR LOGIC IF BIT15 IS SET.

CA READLBIT

MASK FLGWRD11

EXTEND

BZF 35KCHK

# ALT WAS &gt; 35000 FT LAST CYCLE CHK NOW

CAF XORFLBIT

MASK FLGWRD11

EXTEND

BZF XORCHK

# NO -- TEST THIS PASS

HITEST

CAF PSTHIBIT

MASK FLGWRD11

EXTEND

BZF HIGATCHK

# NOT AT HIGATE LAST CYCLE -- CHK THIS CYCLE

POS2CHK

CAF BIT7

EXTEND

RAND CHAN33

# VERIFY LR IN POS2

EXTEND

BZF UPDATCHK

# IT IS -- CHECK FOR LR UPDATE

CAF BIT13

EXTEND

RAND CHAN12

EXTEND

BZF LRPOSALM

# LR NOT IN POS2 OR REPOSITIONING -- BAD

TCF CONTSERV

# LR BEING REPOSITIONED -- CONTINUE SERV

HIGATCHK

CA TTF/8

AD RPCRTIME

EXTEND

BZMF POS1CHK

# NO

CA EBANK4

XCH EBANK

TS L

# MUST SWITCH EBANKS

# SAVE IN L

EBANK= XNBPIP

CS XNBPIP

EBANK= DVCNTR

LXCH EBANK

AD RPCRTQSW

# RESTORE EBANK

# QSW - UXBXP



# SERVICER

PAGE 875

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  |          |        |          |                                            | 2  |
| 3  |          |        |          |                                            | 3  |
| 4  |          | EXTEND |          |                                            | 4  |
| 5  |          | BZMF   | HIGATASK | # IF UXBXP > QSW, THEN REPOSITION          | 5  |
| 6  | POS1CHK  | CAF    | BIT6     | # HIGATE NOT IN SIGHT -- DO POS1 CHK       | 6  |
| 7  |          |        |          |                                            | 7  |
| 8  |          | EXTEND |          |                                            | 8  |
| 9  |          | RAND   | 33       |                                            | 9  |
| 10 |          |        |          |                                            | 10 |
| 11 |          | EXTEND |          |                                            | 11 |
| 12 |          | BZF    | UPDATCHK | # LR IN POS1 -- CHECK FOR LR UPDATE        | 12 |
| 13 | LRPOSALM | TC     | ALARM    | # LR NOT IN PROPER POS-ALARM-BYPASS UPDATE | 13 |
| 14 | CONTSERV | OCT    | 511      | # AND CONTINUE SERVICER                    | 14 |
| 15 |          | INHINT |          |                                            | 15 |
| 16 |          | CS     | BITS4-7  |                                            | 16 |
| 17 |          | MASK   | FLGWRD11 | # CLEAR LR MEASUREMENT MADE DISCRETES.     | 17 |
| 18 |          | TS     | FLGWRD11 |                                            | 18 |
| 19 |          |        |          |                                            | 19 |
| 20 |          | TC     | IBNKCALL | # SET LR LITES PROPERLY                    | 20 |
| 21 |          | CADR   | R12LITES |                                            | 21 |
| 22 |          |        |          |                                            | 22 |
| 23 |          |        |          |                                            | 23 |
| 24 |          |        |          |                                            | 24 |
| 25 |          |        |          |                                            | 25 |
| 26 |          |        |          |                                            | 26 |
| 27 |          |        |          |                                            | 27 |
| 28 |          |        |          |                                            | 28 |
| 29 |          |        |          |                                            | 29 |
| 30 |          |        |          |                                            | 30 |
| 31 |          |        |          |                                            | 31 |
| 32 |          |        |          |                                            | 32 |
| 33 |          |        |          |                                            | 33 |
| 34 |          |        |          |                                            | 34 |
| 35 |          |        |          |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 |          |        |          |                                            | 37 |
| 38 |          |        |          |                                            | 38 |
| 39 |          |        |          |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 |          |        |          |                                            | 41 |
| 42 |          |        |          |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          |        |          |                                            | 44 |
| 45 |          |        |          |                                            | 45 |
| 46 |          |        |          |                                            | 46 |
| 47 |          |        |          |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          |        |          |                                            | 49 |
| 50 |          |        |          |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

1412THE

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| COPYCYC1 | TC     | QUIKFAZ5 |                                            |
| R29?     | CA     | FLAGWRD3 |                                            |
|          | MASK   | NR29&RDR |                                            |
|          | CCS    | A        | # IS NOR29FLG OR READRFLG SET?             |
|          | TCF    | R29NODES | # YES, SO DON'T DESIGNATE.                 |
|          | CA     | RADMODES | # NO, SO R29 IS CALLED FOR.                |
|          | MASK   | OCT10002 | # IS THE RR NOT ZEROING ITS CDUS, AND      |
|          | CCS    | A        | # IS THE RENDEZVOUS RADAR IN AUTO MODE?    |
|          | TCF    | R29NODES | # NO, SO DON'T DESIGNATE.                  |
|          | CA     | RADMODES |                                            |
|          | MASK   | PRI022   |                                            |
|          | CCS    | A        | # IS RR REPOSITIONING OR REMODING?         |
|          | TCF    | NOR29NOW | # YES: COME BACK IN 2 SECONDS & TRY AGAIN. |
|          | TCF    | R29      |                                            |
| R29NODES | INHINT |          | # R29 NOT ALLOWED THIS CYCLE.              |
|          | CS     | DESIGBIT | # SHOW THAT DESIGNATION IS OFF.            |
|          | MASK   | RADMODES |                                            |
|          | TS     | RADMODES |                                            |
| NOR29NOW | TC     | INTPRET  | # INTERPRET DOES A RELINT.                 |
|          | VLOAD  | ABVAL    | # MPAC = ABVAL( NEW SM. POSITION VECTOR )  |
|          | PUSH   | RIS      |                                            |
|          |        | DSU      | #                                          |
|          |        | /LAND/   | (2)                                        |
|          | STORE  | HCALC    | # NEW HCALC*2(24)M.                        |
|          | STORE  | HCALC1   |                                            |
|          | DMPR   | RTB      |                                            |
|          | STOVL  | ALTCONV  |                                            |
|          |        | SGNAGREE |                                            |
|          |        | ALTBITS  | # ALTITUDE FOR R10 IN BIT UNITS.           |
|          | VXV    | UNIT/R/  |                                            |
|          |        | UNIT     |                                            |
|          |        | UHYP     |                                            |
|          | STOVL  | UHYP     | # DOWNRANGE HALF-UNIT VECTOR FOR R10.      |
|          | VXM    | RIS      |                                            |
|          |        | VSR4     |                                            |
|          | STOVL  | REFSMMAT |                                            |
|          |        | RN1      | # TEMP. REF. POSITION VECTOR*2(29)M.       |
|          |        | VIS      |                                            |
|          | VXM    | VSL1     |                                            |
|          |        | REFSMMAT |                                            |
|          | STOVL  | VN1      | # TEMP. REF. VELOCITY VECTOR 2(7) M/CS.    |
|          | VXV    | UNIT/R/  |                                            |
|          |        | ABVAL    |                                            |

|    |          |        |            |                                            |    |
|----|----------|--------|------------|--------------------------------------------|----|
| 1  |          |        |            |                                            | 1  |
| 2  |          |        | V1S        |                                            | 2  |
| 3  |          | SL1    | DSQ        |                                            | 3  |
| 4  |          | DDV    |            |                                            | 4  |
| 5  |          | DMPR   | RTB        |                                            | 5  |
| 6  |          |        | ARCONV1    |                                            | 6  |
| 7  |          |        | SGNAGREE   |                                            | 7  |
| 8  | COPYCYC2 | EXIT   |            | # LEAVE ALTITUDE RATE COMPENSATION IN MPAC | 8  |
| 9  |          | INHINT |            |                                            | 9  |
| 10 |          | CA     | UNIT/R/    | # UPDATE RUNIT FOR R10.                    | 10 |
| 11 |          | TS     | RUNIT      |                                            | 11 |
| 12 |          | CA     | UNIT/R/ +2 |                                            | 12 |
| 13 |          | TS     | RUNIT +1   |                                            | 13 |
| 14 |          | CA     | UNIT/R/ +4 |                                            | 14 |
| 15 |          | TS     | RUNIT +2   |                                            | 15 |
| 16 |          | CA     | MPAC       | # LOAD NEW DALTRATE FOR R10.               | 16 |
| 17 |          | TS     | DALTRATE   |                                            | 17 |
| 18 |          |        |            |                                            | 18 |
| 19 |          | EXTEND |            |                                            | 19 |
| 20 |          | DCA    | R1S        |                                            | 20 |
| 21 |          | DXCH   | R          |                                            | 21 |
| 22 |          | EXTEND |            |                                            | 22 |
| 23 |          | DCA    | R1S +2     |                                            | 23 |
| 24 |          | DXCH   | R +2       |                                            | 24 |
| 25 |          | EXTEND |            |                                            | 25 |
| 26 |          | DCA    | R1S +4     |                                            | 26 |
| 27 |          | DXCH   | R +4       |                                            | 27 |
| 28 |          | EXTEND |            |                                            | 28 |
| 29 |          | DCA    | V1S        |                                            | 29 |
| 30 |          | DXCH   | V          |                                            | 30 |
| 31 |          | EXTEND |            |                                            | 31 |
| 32 |          | DCA    | V1S +2     |                                            | 32 |
| 33 |          | DXCH   | V +2       |                                            | 33 |
| 34 |          | EXTEND |            |                                            | 34 |
| 35 |          | DCA    | V1S +4     |                                            | 35 |
| 36 |          | DXCH   | V +4       |                                            | 36 |
| 37 |          |        |            |                                            | 37 |
| 38 |          | TCF    | COPYCYCL   | # COMPLETE THE COPYCYCL.                   | 38 |
| 39 |          |        |            |                                            | 39 |
| 40 |          |        |            |                                            | 40 |
| 41 |          |        |            |                                            | 41 |
| 42 |          |        |            |                                            | 42 |
| 43 |          |        |            |                                            | 43 |
| 44 |          |        |            |                                            | 44 |
| 45 |          |        |            |                                            | 45 |
| 46 |          |        |            |                                            | 46 |
| 47 |          |        |            |                                            | 47 |
| 48 |          |        |            |                                            | 48 |
| 49 |          |        |            |                                            | 49 |
| 50 |          |        |            |                                            | 50 |
| 51 |          |        |            |                                            | 51 |
| 52 |          |        |            |                                            | 52 |
| 53 |          |        |            |                                            | 53 |
| 54 |          |        |            |                                            | 54 |
| 55 |          |        |            |                                            | 55 |
| 56 |          |        |            |                                            | 56 |
| 57 |          |        |            |                                            | 57 |
| 58 |          |        |            |                                            | 58 |
| 59 |          |        |            |                                            | 59 |
| 60 |          |        |            |                                            | 60 |

# ALTCHK COMPARES CURRENT ALTITUDE (IN HCALC) WITH A SPECIFIED ALTITUDE FROM A TABLE BEGINNING AT ALTCRIT.  
# ITS CALLING SEQUENCE IS AS FOLLOWS:-

|   |     |                                                                   |          |
|---|-----|-------------------------------------------------------------------|----------|
| # | L   | CAF                                                               | N        |
| # | L+1 | TC                                                                | BANKCALL |
| # | L+2 | CADR                                                              | ALTCHK   |
| # | L+3 | RETURN HERE IF HCALC STILL > SPECIFIED CRITERION. C(L) = +0.      |          |
| # | L+4 | RETURN HERE IF HCALC < OR = SPECIFIED CRITERION. C(A) = C(L) = +0 |          |

# ALTCHK MUST BE BANKCALLED EVEN FROM ITS OWN BANK. N IS THE LOCATION, RELATIVE TO THE TAG ALTCRIT,  
# OF THE BEGINNING OF THE DP CONSTANT TO BE USED AS A CRITERION.

|          |        |            |                               |
|----------|--------|------------|-------------------------------|
| ALTCHK   | EXTEND | A          |                               |
|          | INDEX  |            |                               |
|          | DCA    | ALTCRIT    |                               |
|          | DXCH   | MPAC +1    |                               |
|          | EXTEND |            |                               |
|          | DCS    | HCALC      |                               |
|          | DAS    | MPAC +1    |                               |
|          | TC     | BRANCH +4  |                               |
|          | CAF    | ZERO       | # BETTER THAN A NOOP, PERHAPS |
|          | INCR   | BUF2       |                               |
|          | TCF    | SWRETURN   |                               |
| ALTCRIT  | =      | 25KFT      |                               |
| 25KFT    | 2DEC   | 7620 B-24  | # (0)                         |
| 50KFT    | 2DEC   | 15240 B-24 | # (2)                         |
| 50FT     | 2DEC   | 15.24 B-24 | # (4)                         |
| 30KFT    | 2DEC   | 9144 B-24  | # (6)                         |
| 2KFT/SEC | DEC    | 6.096 B-7  | # 2000 FT/SEC AT 2(7) M/CS    |

# (A REMARK WAS LIKELY TO BE NEEDED HERE TO EXPLAIN XORCHK) 4/JUN/09,FB

|        |       |          |                                |
|--------|-------|----------|--------------------------------|
| XORCHK | CAF   | SIX      | # ARE WE BELOW 30000 FT?       |
|        | TC    | BANKCALL |                                |
|        | CADR  | ALTCHK   |                                |
|        | TCF   | HITEST   | # CONTINUE LR UPDATE           |
|        | TC    | UPFLAG   | # YES: INHIBIT X-AXIS OVERRIDE |
|        | ADRES | XOVINFLG |                                |
|        | TC    | UPFLAG   |                                |
|        | ADRES | XORFLG   |                                |
|        | TCF   | HITEST   | # CONTINUE LR UPDATE           |
| 35KCHK | CAF   | TWO      | # ARE WE BELOW 35000 FT?       |



# SERVICER

PAGE 879

1412THE

|    |  |       |          |    |
|----|--|-------|----------|----|
| 1  |  |       |          | 1  |
| 2  |  | TC    | BANKCALL | 2  |
| 3  |  | CADR  | ALTCHK   | 3  |
| 4  |  | TCF   | CONTSERV | 4  |
| 5  |  | TC    | UPFLAG   | 5  |
| 6  |  | ADRES | READLR   | 6  |
| 7  |  | TCF   | CONTSERV | 7  |
| 8  |  |       |          | 8  |
| 9  |  |       |          | 9  |
| 10 |  |       |          | 10 |
| 11 |  |       |          | 11 |
| 12 |  |       |          | 12 |
| 13 |  |       |          | 13 |
| 14 |  |       |          | 14 |
| 15 |  |       |          | 15 |
| 16 |  |       |          | 16 |
| 17 |  |       |          | 17 |
| 18 |  |       |          | 18 |
| 19 |  |       |          | 19 |
| 20 |  |       |          | 20 |
| 21 |  |       |          | 21 |
| 22 |  |       |          | 22 |
| 23 |  |       |          | 23 |
| 24 |  |       |          | 24 |
| 25 |  |       |          | 25 |
| 26 |  |       |          | 26 |
| 27 |  |       |          | 27 |
| 28 |  |       |          | 28 |
| 29 |  |       |          | 29 |
| 30 |  |       |          | 30 |
| 31 |  |       |          | 31 |
| 32 |  |       |          | 32 |
| 33 |  |       |          | 33 |
| 34 |  |       |          | 34 |
| 35 |  |       |          | 35 |
| 36 |  |       |          | 36 |
| 37 |  |       |          | 37 |
| 38 |  |       |          | 38 |
| 39 |  |       |          | 39 |
| 40 |  |       |          | 40 |
| 41 |  |       |          | 41 |
| 42 |  |       |          | 42 |
| 43 |  |       |          | 43 |
| 44 |  |       |          | 44 |
| 45 |  |       |          | 45 |
| 46 |  |       |          | 46 |
| 47 |  |       |          | 47 |
| 48 |  |       |          | 48 |
| 49 |  |       |          | 49 |
| 50 |  |       |          | 50 |
| 51 |  |       |          | 51 |
| 52 |  |       |          | 52 |
| 53 |  |       |          | 53 |
| 54 |  |       |          | 54 |
| 55 |  |       |          | 55 |
| 56 |  |       |          | 56 |
| 57 |  |       |          | 57 |
| 58 |  |       |          | 58 |
| 59 |  |       |          | 59 |
| 60 |  |       |          | 60 |

# SET READLR FLAG TO ENABLE LR READING.

# \*\*\*\*\*

|         |       |         |                                         |
|---------|-------|---------|-----------------------------------------|
| CALCGRV | UNIT  | PUSH    | # SAVE UNIT/R/ IN PUSHLIST (18)         |
|         | STORE | UNIT/R/ |                                         |
|         | LXC,1 | SLOAD   | # RTX2 = 0 IF EARTH ORBIT, =2 IF LUNAR. |

|  |       |      |
|--|-------|------|
|  | DCOMP | RTX2 |
|  |       | RTX2 |
|  |       | BMN  |

|  |       |          |        |
|--|-------|----------|--------|
|  | VLOAD | CALCGRV1 |        |
|  |       | DOT      | # (12) |
|  |       | UNITZ    |        |

|  |     |         |        |
|--|-----|---------|--------|
|  | SL1 | UNIT/R/ |        |
|  | DSQ | PUSH    | # (14) |
|  |     | BDSU    |        |

|  |      |        |
|--|------|--------|
|  | PDDL | DP1/20 |
|  |      | DDV    |
|  |      | RESQ   |

|  |       |     |                      |
|--|-------|-----|----------------------|
|  | STORE | 34D | # (RN)SQ             |
|  | DMP   | 32D | # TEMP FOR (RE/RN)SQ |
|  |       | DMP |                      |

|  |      |         |
|--|------|---------|
|  | VXSC | 20J     |
|  |      | PDDL    |
|  |      | UNIT/R/ |

|  |     |     |
|--|-----|-----|
|  | DMP | DMP |
|  |     | 2J  |
|  |     | 32D |

|  |      |       |
|--|------|-------|
|  | VXSC | VSL1  |
|  |      | UNITZ |
|  | VAD  | STADR |

|          |       |          |                                        |
|----------|-------|----------|----------------------------------------|
| CALCGRV1 | STORE | UNITGOBL |                                        |
|          | VAD   | PUSH     | # MPAC = UNIT GRAVITY VECTOR. (18)     |
|          | DLOAD | NORM     | # PERFORM A NORMALIZATION ON RMAGSQ IN |
|          |       | 34D      | # ORDER TO BE ABLE TO SCALE THE MU FOR |
|          |       | X2       | # MAXIMUM PRECISION.                   |

|  |       |      |
|--|-------|------|
|  | BDDV* | SLR* |
|--|-------|------|

|  |      |          |
|--|------|----------|
|  |      | -MUDT,1  |
|  |      | 0 -21D,2 |
|  | VXSC | STADR    |

|  |       |        |                        |
|--|-------|--------|------------------------|
|  | STORE | GDT1/2 | # SCALED AT 2(+7) M/CS |
|  | RVQ   |        |                        |

|         |       |          |
|---------|-------|----------|
| CALCRVG | VLOAD | VXM      |
|         |       | DELV     |
|         |       | REFSMMAT |

|  |       |         |
|--|-------|---------|
|  | VXSC  | VSL1    |
|  |       | KPIP1   |
|  | STORE | DELVREF |

|  |      |      |                                             |
|--|------|------|---------------------------------------------|
|  | VSR1 | PUSH |                                             |
|  | VAD  | PUSH | # (DV-OLDGDT)/2 TO PD SCALED AT 2(+7) M/CS. |



# SERVICER

PAGE 881

|    |         |          |                                        |    |
|----|---------|----------|----------------------------------------|----|
| 1  |         |          |                                        | 1  |
| 2  |         |          |                                        | 2  |
| 3  | VAD     | GDT/2    |                                        | 3  |
| 4  |         | PDDL     |                                        | 4  |
| 5  |         | VN       |                                        | 5  |
| 6  | SL      | PGUIDE   |                                        | 6  |
| 7  |         | VXSC     |                                        | 7  |
| 8  | VAD     | 6D       |                                        | 8  |
| 9  |         | STQ      |                                        | 9  |
| 10 |         | RN       |                                        | 10 |
| 11 | STCALL  | 31D      | # TEMP STORAGE OF RN SCALED 2(+29) M   | 11 |
| 12 |         | RN1      |                                        | 12 |
| 13 |         | CALCGRAV |                                        | 13 |
| 14 | VAD     | VAD      |                                        | 14 |
| 15 | VAD     |          |                                        | 15 |
| 16 |         | VN       |                                        | 16 |
| 17 | STCALL  | VN1      | # TEMP STORAGE OF VN SCALED 2(+7) M/CS | 17 |
| 18 |         | 31D      |                                        | 18 |
| 19 |         |          |                                        | 19 |
| 20 | DP1/20  | 2DEC     | 0.05                                   | 20 |
| 21 | SHIFT11 | 2DEC     | 1 B-11                                 | 21 |
| 22 |         |          |                                        | 22 |
| 23 |         |          |                                        | 23 |
| 24 |         |          |                                        | 24 |
| 25 |         |          |                                        | 25 |
| 26 |         |          |                                        | 26 |
| 27 |         |          |                                        | 27 |
| 28 |         |          |                                        | 28 |
| 29 |         |          |                                        | 29 |
| 30 |         |          |                                        | 30 |
| 31 |         |          |                                        | 31 |
| 32 |         |          |                                        | 32 |
| 33 |         |          |                                        | 33 |
| 34 |         |          |                                        | 34 |
| 35 |         |          |                                        | 35 |
| 36 |         |          |                                        | 36 |
| 37 |         |          |                                        | 37 |
| 38 |         |          |                                        | 38 |
| 39 |         |          |                                        | 39 |
| 40 |         |          |                                        | 40 |
| 41 |         |          |                                        | 41 |
| 42 |         |          |                                        | 42 |
| 43 |         |          |                                        | 43 |
| 44 |         |          |                                        | 44 |
| 45 |         |          |                                        | 45 |
| 46 |         |          |                                        | 46 |
| 47 |         |          |                                        | 47 |
| 48 |         |          |                                        | 48 |
| 49 |         |          |                                        | 49 |
| 50 |         |          |                                        | 50 |
| 51 |         |          |                                        | 51 |
| 52 |         |          |                                        | 52 |
| 53 |         |          |                                        | 53 |
| 54 |         |          |                                        | 54 |
| 55 |         |          |                                        | 55 |
| 56 |         |          |                                        | 56 |
| 57 |         |          |                                        | 57 |
| 58 |         |          |                                        | 58 |
| 59 |         |          |                                        | 59 |
| 60 |         |          |                                        | 60 |

1412THE



\*\*\*\*\*

# MUNRVG IS A SPECIAL AVERAGE G INTEGRATION ROUTINE USED BY THRUSTING  
# PROGRAMS WHICH FUNCTION IN THE VICINITY OF AN ASSUMED SPHERICAL MOON.  
# THE INPUT AND OUTPUT QUANTITIES ARE REFERENCED TO THE STABLE MEMBER  
# COORDINATE SYSTEM.

RVBOTH

VLOAD

PUSH  
G(CSM)

VAD

PDDL  
V(CSM)  
PGUIDE

DDV

VXSC  
SHIFT11

VAD

STCALL

R(CSM)  
R1S  
MUNGRAV

VAD

VAD  
V(CSM)

STADR

STORE

V1S

EXIT

TC

QUIKFAZ5

TC

INTPRET

VLOAD

GDT1/2

STOVL

G(CSM)  
R1S

STOVL

R(CSM)  
V1S

STORE

V(CSM)

EXIT

TC

QUIKFAZ5

TC

INTPRET

VLOAD

VXSC

MUNRVG

DELV

KPIP2

PUSH

VAD

# 1ST PUSH: DELV IN UNITS OF 2(8) M/CS

GDT/2

PUSH

VAD

# 2ND PUSH: (DELV + GDT)/2, UNITS OF 2(7)  
# (12)

V

PDDL

DDV

PGUIDE

SHIFT11

VXSC

VAD

R

STCALL

R1S

# STORE R SCALED AT 2(+24) M

MUNGRAV

|    |           |        |                                             |    |
|----|-----------|--------|---------------------------------------------|----|
| 1  | # SERVICE |        |                                             | 1  |
| 2  |           | VAD    | VAD                                         | 2  |
| 3  |           | VAD    |                                             | 3  |
| 4  |           |        | V                                           | 4  |
| 5  |           | STORE  | V1S                                         | 5  |
| 6  |           | ABVAL  | # STORE V SCALED AT 2(+7) M/CS.             | 6  |
| 7  |           | STOVL  | ABVEL                                       | 7  |
| 8  |           |        | UNIT/R/                                     | 8  |
| 9  |           | DOT    | SL1                                         | 9  |
| 10 |           |        | V1S                                         | 10 |
| 11 |           | STOVL  | HDOTDISP                                    | 11 |
| 12 |           |        | R1S                                         | 12 |
| 13 |           |        | # HDOT = V. UNIT(R)*2(7) M/CS.              | 13 |
| 14 |           | VXV    | VSL2                                        | 14 |
| 15 |           | STODL  | WM                                          | 15 |
| 16 |           |        | DELVS                                       | 16 |
| 17 |           |        | # LUNAR ROTATION CORRECTION TERM*2(5) M/CS. | 17 |
| 18 |           | DSU    | 36D                                         | 18 |
| 19 |           |        | /LAND/                                      | 19 |
| 20 |           | STCALL | HCALC                                       | 20 |
| 21 |           |        | MUNRETRN                                    | 21 |
| 22 | MUNGRAV   | UNIT   | # FOR NOW, DISPLAY WHETHER POS OR NEG       | 22 |
| 23 |           | STODL  | # AT 36D HAVE ABVAL(R), AT 34D R.R          | 23 |
| 24 |           | SL     | UNIT/R/                                     | 24 |
| 25 |           |        | 34D                                         | 25 |
| 26 |           |        | BDDV                                        | 26 |
| 27 |           | DMP    | 6D                                          | 27 |
| 28 |           |        | -MUDTMUN                                    | 28 |
| 29 |           |        | VXSC                                        | 29 |
| 30 |           | STORE  | SHIFT11                                     | 30 |
| 31 |           | RVQ    | UNIT/R/                                     | 31 |
| 32 |           |        | GDT1/2                                      | 32 |
| 33 | 1.95SECS  | DEC    | # 1/2GDT SCALED AT 2(7) M/CS.               | 33 |
| 34 | 7.5       | 2DEC   |                                             | 34 |
| 35 | 2SEC(18)  | 2DEC   | 195                                         | 35 |
| 36 | 2SEC(28)  | 2OCT   | .02286 B-6                                  | 36 |
| 37 | 4SEC(28)  | 2DEC   | # 7.5 FT/SEC AT 2(6) M/CS                   | 37 |
| 38 | BITS4-7   | OCT    | 200 B-18                                    | 38 |
| 39 |           |        | 0000000310                                  | 39 |
| 40 |           |        | # 2SEC AT 2(28)                             | 40 |
| 41 |           |        |                                             | 41 |
| 42 |           |        |                                             | 42 |
| 43 |           |        |                                             | 43 |
| 44 |           |        |                                             | 44 |
| 45 |           |        |                                             | 45 |
| 46 |           |        |                                             | 46 |
| 47 |           |        |                                             | 47 |
| 48 |           |        |                                             | 48 |
| 49 |           |        |                                             | 49 |
| 50 |           |        |                                             | 50 |
| 51 |           |        |                                             | 51 |
| 52 |           |        |                                             | 52 |
| 53 |           |        |                                             | 53 |
| 54 |           |        |                                             | 54 |
| 55 |           |        |                                             | 55 |
| 56 |           |        |                                             | 56 |
| 57 |           |        |                                             | 57 |
| 58 |           |        |                                             | 58 |
| 59 |           |        |                                             | 59 |
| 60 |           |        |                                             | 60 |

|    |          |        |          |                                           |
|----|----------|--------|----------|-------------------------------------------|
| 1  | UPDATCHK | CAF    | NOLRRBIT | # SEE IF LR UPDATE INHIBITED.             |
| 2  |          | MASK   | FLGWRD11 |                                           |
| 3  |          | CCS    | A        |                                           |
| 4  |          | TCF    | CONTSERV | # IT IS -- NO LR UPDATE                   |
| 5  |          | CAF    | RNGEDBIT | # NO INHIBIT -- SEE ALT MEAS. THIS CYCLE. |
| 6  |          | MASK   | FLGWRD11 |                                           |
| 7  |          | EXTEND |          |                                           |
| 8  |          | BZF    | VMEASCHK | # NO ALT MEAS THIS CYCLE -- CHECK FOR VEL |
| 9  |          |        |          |                                           |
| 10 | POSUPDAT | CA     | FIXLOC   | # SET PUSHLIST TO ZERO                    |
| 11 |          | TS     | PUSHLOC  |                                           |
| 12 |          |        |          |                                           |
| 13 |          | TC     | INTPRET  |                                           |
| 14 |          | VLOAD  | VXM      |                                           |
| 15 |          |        | HBEAMNB  |                                           |
| 16 |          |        | XNBPIP   | # HBEAM SM AT 2(2)                        |
| 17 |          | PDVL   | VSL2     | # STORE HBEAM IN PD 0-5                   |
| 18 |          |        | VIS      | # SCALE V AT 2(5) M/CS                    |
| 19 |          | VAD    | DOT      |                                           |
| 20 |          |        | DELVS    | # V RELATIVE TO SURFACE AT 2(5) M/CS      |
| 21 |          |        | OD       | # V ALONG HBEAM AT 2(7) M/CS.             |
| 22 |          | DMP    | EXIT     |                                           |
| 23 |          |        | RADSKAL  | # SCALE TO RADAR COUNTS X 5               |
| 24 |          |        |          |                                           |
| 25 |          | CS     | FLGWRD12 | # TEST LR ALTITUDE SCALE FACTOR           |
| 26 |          | MASK   | ALTSCBIT |                                           |
| 27 |          | EXTEND |          |                                           |
| 28 |          | BZF    | +3       | # BRANCH IF HIGH SCALE                    |
| 29 |          |        |          |                                           |
| 30 |          | CA     | SKALSKAL | # RESCALE IF LOW SCALE                    |
| 31 |          | TC     | SHORTMP  |                                           |
| 32 |          |        |          |                                           |
| 33 |          |        |          |                                           |
| 34 | +3       | TC     | INTPRET  |                                           |
| 35 |          | DAD    | SL       | # CORRECT HMEAS FOR DOPPLER EFFECT        |
| 36 |          |        | HMEAS    |                                           |
| 37 |          |        | 7D       |                                           |
| 38 |          | DMP    | VXSC     | # SLANT RANGE AT 2(21), PUSH UP FOR HBEAM |
| 39 |          |        | HSCAL    | # SLANT RANGE VECTOR AT 2(23) M           |
| 40 |          | DOT    | DSU      |                                           |
| 41 |          |        | UNIT/R/  | # ALTITUDE AT 2(24) M                     |
| 42 |          |        | HCALC    | # DELTA H AT 2(24) M                      |
| 43 |          | STORE  | DELTAH   |                                           |
| 44 |          | EXIT   |          |                                           |
| 45 |          |        |          |                                           |
| 46 |          | CA     | FLGWRD11 |                                           |
| 47 |          | MASK   | PSTHIBIT |                                           |
| 48 |          | EXTEND |          | # DO NOT PERFORM DATA REASONABLENESS TEST |
| 49 |          | BZF    | NOREASON | # UNTIL AFTER HIGATE                      |
| 50 |          |        |          |                                           |
| 51 |          |        |          |                                           |
| 52 |          |        |          |                                           |
| 53 |          |        |          |                                           |
| 54 |          |        |          |                                           |
| 55 |          |        |          |                                           |
| 56 |          |        |          |                                           |
| 57 |          |        |          |                                           |
| 58 |          |        |          |                                           |
| 59 |          |        |          |                                           |
| 60 |          |        |          |                                           |

|    |          |        |          |                                            |
|----|----------|--------|----------|--------------------------------------------|
| 1  |          |        |          |                                            |
| 2  |          | TC     | INTPRET  |                                            |
| 3  |          | ABS    | DSU      |                                            |
| 4  |          |        | DELQFIX  | # ABS(DELTAH) - DQFIX 50 FT NOM            |
| 5  |          | SL3    | DSU      | # SCALE TO 2(21)                           |
| 6  |          |        | HCALC    | # ABS(DELTAH) - (50 + HCALC/8) AT 2(21)    |
| 7  |          | EXIT   |          |                                            |
| 8  |          |        |          |                                            |
| 9  |          | INCR   | LRLCTR   |                                            |
| 10 |          | TC     | BRANCH   |                                            |
| 11 |          | TCF    | HFAIL    | # DELTA H TOO LARGE                        |
| 12 |          | TCF    | HFAIL    | # DELTA H TOO LARGE                        |
| 13 |          | TC     | DOWNFLAG | # TURN OFF ALT FAIL LAMP                   |
| 14 |          | ADRES  | HFLSHFLG |                                            |
| 15 |          |        |          |                                            |
| 16 | NOREASON | CS     | FLGWRD11 |                                            |
| 17 |          | MASK   | LRINHBIT |                                            |
| 18 |          | CCS    | A        |                                            |
| 19 |          | TCF    | VMEASCHK | # UPDATE INHIBITED -- TEST VELOCITY ANYWAY |
| 20 |          |        |          |                                            |
| 21 |          | TC     | INTPRET  | # DO POSITION UPDATE                       |
| 22 |          | DLOAD  | SR4      |                                            |
| 23 |          |        | HCALC    | # RESCALE H TO 2(28)M                      |
| 24 |          | EXIT   |          |                                            |
| 25 |          | EXTEND |          |                                            |
| 26 |          | DCA    | DELTAH   | # STORE DELTAH IN MPAC AND                 |
| 27 |          | DXCH   | MPAC     | # BRING HCALC INTO A,L                     |
| 28 |          | TC     | ALSIGNAG |                                            |
| 29 |          | EXTEND |          | # IF HIGH PART OF HCALC IS NON-ZERO, THEN  |
| 30 |          | BZF    | +2       | # HCALC > HMAX,                            |
| 31 |          | TCF    | VMEASCHK | # SO UPDATE IS BYPASSED                    |
| 32 |          | TS     | MPAC +2  | # FOR LATER SHORTMP                        |
| 33 |          |        |          |                                            |
| 34 |          | CS     | L        | # -H AT 2(14) M                            |
| 35 |          | AD     | LRHMAX   | # HMAX - H                                 |
| 36 |          | EXTEND |          |                                            |
| 37 |          | BZMF   | VMEASCHK | # IF H >HMAX, BYPASS UPDATE                |
| 38 |          | EXTEND |          |                                            |
| 39 |          | MP     | LRWH     | # WH(HMAX - H)                             |
| 40 |          | EXTEND |          |                                            |
| 41 |          | DV     | LRHMAX   | # WH(1 - H/HMAX)                           |
| 42 |          | TS     | MPTMP    |                                            |
| 43 |          | TC     | SHORTMP2 | # DELTAH (WH)(1 - H/HMAX) IN MPAC          |
| 44 |          | TC     | INTPRET  | # MODE IS DP FROM ABOVE                    |
| 45 |          | SL1    |          |                                            |
| 46 |          | VXSC   | VAD      |                                            |
| 47 |          |        | UNIT/R/  | # DELTAR = DH(WH)(1 - H/HMAX) UNIT/R/      |
| 48 |          |        | R1S      |                                            |
| 49 |          | STCALL | GNUR     |                                            |
| 50 |          |        | MUNGRAV  |                                            |
| 51 |          | EXIT   |          |                                            |
| 52 |          |        |          |                                            |
| 53 |          |        |          |                                            |
| 54 |          |        |          |                                            |
| 55 |          |        |          |                                            |
| 56 |          |        |          |                                            |
| 57 |          |        |          |                                            |
| 58 |          |        |          |                                            |
| 59 |          |        |          |                                            |
| 60 |          |        |          |                                            |

# SERVICER

|    |          |        |            |                                    |
|----|----------|--------|------------|------------------------------------|
| 1  |          |        |            |                                    |
| 2  |          | TC     | QUIKFAZ5   |                                    |
| 3  |          |        |            |                                    |
| 4  |          | CA     | ZERO       |                                    |
| 5  | RUPDATED | TC     | GNURVST    |                                    |
| 6  |          |        |            |                                    |
| 7  | VMEASCHK | TC     | QUIKFAZ5   | # RESTART AT NEXT LOCATION         |
| 8  |          | CS     | FLGWRD11   |                                    |
| 9  |          | MASK   | VELDABIT   | # IS V READING AVAILABLE?          |
| 10 |          | CCS    | A          |                                    |
| 11 |          | TCF    | VALTCHK    | # NO: SEE IF V READING TO BE TAKEN |
| 12 |          |        |            |                                    |
| 13 | VELUPDAT | CS     | VSELECT    | # PROCESS VELOCITY DATA            |
| 14 |          | TS     | L          |                                    |
| 15 |          | ADS    | L          | # -2 VSELECT IN L                  |
| 16 |          | AD     | L          |                                    |
| 17 |          | AD     | L          | # -6 VSELECT IN A                  |
| 18 |          | INDEX  | FIXLOC     |                                    |
| 19 |          | DXCH   | X1         | # X1 = -6 VSELECT, X2 = -2 VSELECT |
| 20 |          |        |            |                                    |
| 21 |          | CA     | EBANK4     |                                    |
| 22 |          | TS     | EBANK      |                                    |
| 23 |          | EBANK= | LRXCDU     |                                    |
| 24 |          |        |            |                                    |
| 25 |          | CA     | LRYCDU     | # STORE LRCDUS IN CDUSPOTS         |
| 26 |          | TS     | CDUSPOT    |                                    |
| 27 |          | CA     | LRZCDU     |                                    |
| 28 |          | TS     | CDUSPOT +2 |                                    |
| 29 |          | CA     | LRXCDU     |                                    |
| 30 |          | TS     | CDUSPOT +4 |                                    |
| 31 |          |        |            |                                    |
| 32 |          | TC     | BANKCALL   |                                    |
| 33 |          | CADR   | QUICTRIG   | # GET SINES AND COSINES FOR NBSM   |
| 34 |          |        |            |                                    |
| 35 |          | CA     | FIXLOC     |                                    |
| 36 |          | TS     | PUSHLOC    | # SET PD TO ZERO                   |
| 37 |          |        |            |                                    |
| 38 |          | TC     | INTPRET    |                                    |
| 39 |          | VLOAD* | CALL       |                                    |
| 40 |          |        | VZBEAMNB,1 | # CONVERT VBEAM FROM NB TO SM      |
| 41 |          |        | *NBSM*     |                                    |
| 42 |          | PDDL   | SL         | # STORE IN PD 0-5                  |
| 43 |          |        | VMEAS      | # LOAD VELOCITY MEASUREMENT        |
| 44 |          |        | 12D        |                                    |
| 45 |          | DMP*   | PUSH       | # SCALE TO M/CS AT 2(6)            |
| 46 |          |        | VZSCAL,2   | # AND STORE IN PD 6-7              |
| 47 |          | EXIT   |            |                                    |
| 48 |          | CS     | ONE        |                                    |
| 49 |          | TS     | MODE       | # CHANGE STORE MODE TO VECTOR      |
| 50 |          |        |            |                                    |
| 51 |          | CA     | PIPTM      | # STORE DELV IN MPAC               |
| 52 |          |        |            |                                    |
| 53 |          |        |            |                                    |
| 54 |          |        |            |                                    |
| 55 |          |        |            |                                    |
| 56 |          |        |            |                                    |
| 57 |          |        |            |                                    |
| 58 |          |        |            |                                    |
| 59 |          |        |            |                                    |
| 60 |          |        |            |                                    |

|        |           |                                          |
|--------|-----------|------------------------------------------|
| ZL     |           |                                          |
| DXCH   | MPAC      |                                          |
| CA     | PIPTEM +1 |                                          |
| ZL     |           |                                          |
| DXCH   | MPAC +3   |                                          |
| CA     | PIPTEM +2 |                                          |
| ZL     |           |                                          |
| DXCH   | MPAC +5   |                                          |
| CA     | EBANK7    |                                          |
| TS     | EBANK     | # RESTORE EBANK 7                        |
| EBANK= | DVCNTR    |                                          |
| TC     | INTPRET   |                                          |
| VXSC   | PDDL      |                                          |
|        | KPIP1     | # SCALE DELV TO 2(7) M/CS AND PUSH       |
|        | LRVTIME   | # TIME OF DELV AT 2(28) CS               |
| DSU    | DDV       |                                          |
|        | PIPTIME   | # TU - T(N-1)                            |
|        | 2SEC(28)  |                                          |
| VXSC   | VSL1      | # G(N-1)(TU - T(N-1))                    |
|        | GDT/2     | # SCALED AT 2(7) M/CS                    |
| VAD    | VAD       | # PUSH UP FOR DELV                       |
|        | V         | # VU = V(N-1) + DELVU + G(N-1) DTU       |
| VSL2   | VAD       | # SCALE TO 2(5) M/CS AND SUBTRACT        |
|        | DELVS     | # MOON ROTATION.                         |
| PUSH   | ABVAL     | # STORE IN PD                            |
| SR4    | DAD       | # ABS(VM)/8 + 7.5 AT 2(6)                |
|        | 7.5       |                                          |
| STOVL  | 20D       | # STORE IN 20D AND PICK UP VM            |
| DOT    | BDSU      | # V(EST) AT 2(6)                         |
|        | 0         | # DELTAV = VMEAS - V(EST)                |
| PUSH   | ABS       |                                          |
| DSU    | EXIT      | # ABS(DV) - (7.5 + ABS(VM)/8))           |
|        | 20D       |                                          |
| INCR   | LRMCTR    |                                          |
| TC     | BRANCH    |                                          |
| TCF    | VFAIL     | # DELTA V TOO LARGE. ALARM               |
| TCF    | VFAIL     | # DELTA V TOO LARGE. ALARM               |
| TC     | DOWNFLAG  | # TURN OFF VEL FAIL LAMP                 |
| ADRES  | VFLSHFLG  |                                          |
| CA     | FLGWRD11  |                                          |
| MASK   | VXINHBIT  |                                          |
| EXTEND |           |                                          |
| BZF    | VUPDAT    | # IF VX INHIBIT RESET, INCORPORATE DATA. |

# SERVICER

|    |        |        |          |                                      |  |
|----|--------|--------|----------|--------------------------------------|--|
| 1  |        |        |          |                                      |  |
| 2  |        | TC     | DOWNFLAG |                                      |  |
| 3  |        | ADRES  | VXINH    | # RESET VX INHIBIT                   |  |
| 4  |        |        |          |                                      |  |
| 5  |        | CA     | VSELECT  |                                      |  |
| 6  |        | AD     | NEG2     | # IF VSELECT = 2 (X AXIS).           |  |
| 7  |        | EXTEND |          | # BYPASS UPDATE                      |  |
| 8  |        | BZF    | ENDVDAT  |                                      |  |
| 9  |        |        |          |                                      |  |
| 10 | VUPDAT | CS     | FLGWRD11 |                                      |  |
| 11 |        | MASK   | LRINHBIT |                                      |  |
| 12 |        | CCS    | A        |                                      |  |
| 13 |        | TCF    | VALTCHK  | # UPDATE INHIBITED                   |  |
| 14 |        |        |          |                                      |  |
| 15 |        | TS     | MPAC +1  |                                      |  |
| 16 |        |        |          |                                      |  |
| 17 |        | CA     | ABVEL    | # STORE E7 ERASABLES NEEDED IN TEMPS |  |
| 18 |        | TS     | ABVEL*   |                                      |  |
| 19 |        | CA     | VSELECT  |                                      |  |
| 20 |        | TS     | VSELECT* |                                      |  |
| 21 |        | CA     | EBANK5   |                                      |  |
| 22 |        | TS     | EBANK    | # CHANGE EBANKS                      |  |
| 23 |        |        |          |                                      |  |
| 24 |        | EBANK= | LRVF     |                                      |  |
| 25 |        | CS     | LRVF     |                                      |  |
| 26 |        | AD     | ABVEL*   | # IF V < VF, USE WVF                 |  |
| 27 |        | EXTEND |          |                                      |  |
| 28 |        | BZMF   | USEVF    |                                      |  |
| 29 |        |        |          |                                      |  |
| 30 |        | CS     | ABVEL*   |                                      |  |
| 31 |        | AD     | LRVMAX   | # VMAX - V                           |  |
| 32 |        | EXTEND |          |                                      |  |
| 33 |        | BZMF   | WSTOR -1 | # IF V > VMAX, W = 0                 |  |
| 34 |        |        |          |                                      |  |
| 35 |        | EXTEND |          |                                      |  |
| 36 |        | INDEX  | VSELECT* |                                      |  |
| 37 |        | MP     | LRWVZ    | # WV(VMAX - V)                       |  |
| 38 |        |        |          |                                      |  |
| 39 |        | EXTEND |          |                                      |  |
| 40 |        | DV     | LRVMAX   | # WV( 1 - V/VMAX )                   |  |
| 41 |        | TCF    | WSTOR    |                                      |  |
| 42 |        |        |          |                                      |  |
| 43 | USEVF  | INDEX  | VSELECT* |                                      |  |
| 44 |        | CA     | LRWVFZ   | # USE APPROPRIATE CONSTANT WEIGHT    |  |
| 45 |        | TCF    | WSTOR    |                                      |  |
| 46 |        |        |          |                                      |  |
| 47 | -1     | CA     | ZERO     |                                      |  |
| 48 | WSTOR  | TS     | MPAC     |                                      |  |
| 49 |        | CS     | BIT7     | # (=64D)                             |  |
| 50 |        | AD     | MODREG   |                                      |  |
| 51 |        | EXTEND |          |                                      |  |
| 52 |        |        |          |                                      |  |
| 53 |        |        |          |                                      |  |
| 54 |        |        |          |                                      |  |
| 55 |        |        |          |                                      |  |
| 56 |        |        |          |                                      |  |
| 57 |        |        |          |                                      |  |
| 58 |        |        |          |                                      |  |
| 59 |        |        |          |                                      |  |
| 60 |        |        |          |                                      |  |

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
|          | BZMF   | +3       | # IF IN P65,P66,P67, USE ANOTHER CONSTANT |
|          | CA     | LRWVFF   |                                           |
|          | TS     | MPAC     |                                           |
| +3       | CA     | EBANK7   |                                           |
|          | TS     | EBANK    | # CHANGE EBANKS                           |
|          | EBANK= | ABVEL    |                                           |
|          | TC     | INTPRET  |                                           |
|          | DMP    | VXSC     | # W(DELTA V)(VBEAMSM) UP 6-7, 0-5         |
|          | VAD    |          |                                           |
|          |        | V1S      | # ADD WEIGHTED DELTA V TO VELOCITY        |
|          | STORE  | GNUV     |                                           |
|          | EXIT   |          |                                           |
|          | TC     | QUIKFAZ5 | # DO NOT RE-UPDATE                        |
|          | CA     | SIX      |                                           |
| VUPDATED | TC     | GNURVST  | # STORE NEW VELOCITY VECTOR               |
| ENDVDAT  | =      | VALTCHK  |                                           |
| VALTCHK  | TC     | QUIKFAZ5 | # DO NOT REPEAT ABOVE                     |
|          | CAF    | READVBIT | # TEST READVEL TO SEE IF VELOCITY READING |
|          | MASK   | FLGWRD11 | # IS DESIRED.                             |
|          | CCS    | A        |                                           |
|          | TCF    | READV    | # YES -- READ VELOCITY                    |
|          | CS     | ABVEL    | # NO -- SEE IF VELOCITY < 2000 FT/SEC     |
|          | AD     | 2KFT/SEC |                                           |
|          | EXTEND |          |                                           |
|          | BZMF   | CONTSERV | # V > 2000 FT/SEC DO NOT READ VEL         |
|          | TC     | UPFLAG   | # V < 2000 FT/SEC SET READVEL AND READ.   |
|          | ADRES  | READVEL  |                                           |
| READV    | CAF    | PRI032   | # SET UP JOB TO READ VELOCITY BEAMS.      |
|          | TC     | NOVAC    |                                           |
|          | EBANK= | HMEAS    |                                           |
|          | 2CADR  | LRVJOB   |                                           |
|          | TCF    | CONTSERV | # CONTINUE WITH SERVICER                  |
| GNURVST  | TS     | BUF      | # STORE GNUR (=GNUV) IN R1S OR V1S        |
|          | EXTEND |          | # A = 0 FOR R, A = 6 FOR V                |
|          | DCA    | GNUR     |                                           |
|          | INDEX  | BUF      |                                           |
|          | DXCH   | R1S      |                                           |
|          | EXTEND |          |                                           |



|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
|          | DCA    | GNUR +2  |                                            |
|          | INDEX  | BUF      |                                            |
|          | DXCH   | RIS +2   |                                            |
|          | EXTEND |          |                                            |
|          | DCA    | GNUR +4  |                                            |
|          | INDEX  | BUF      |                                            |
|          | DXCH   | RIS +4   |                                            |
|          | TC     | Q        |                                            |
| QUIKFAZ5 | CA     | EBANK3   |                                            |
|          | XCH    | EBANK    | # SET EBANK 3                              |
|          | DXCH   | L        | # Q TO A, A TO L                           |
|          | EBANK= | PHSNAME5 |                                            |
|          | TS     | PHSNAME5 |                                            |
|          | LXCH   | EBANK    |                                            |
|          | EBANK= | DVCNTR   |                                            |
|          | TC     | A        |                                            |
| HFAIL    | CS     | LRRCTR   |                                            |
|          | EXTEND |          |                                            |
|          | BZF    | NORLITE  | # IF R = 0, DO NOT TURN ON TRK FAIL        |
|          | AD     | LRLCTR   |                                            |
|          | MASK   | NEG3     |                                            |
|          | EXTEND |          | # IF L-R LT 4, DO NOT TURN ON TRK FAIL     |
|          | BZF    | +2       |                                            |
|          | TCF    | NORLITE  |                                            |
|          | TC     | UPFLAG   | # AND SET BIT TO TURN ON TRACKER FAIL LITE |
|          | ADRES  | HFLSHFLG |                                            |
| NORLITE  | CA     | LRLCTR   |                                            |
|          | TS     | LRRCTR   | # SET R = L                                |
|          | TCF    | VMEASCHK |                                            |
| VFAIL    | CS     | LRSCCTR  | # DELTA Q LARGE                            |
|          | EXTEND |          | # IF S = 0, DO NOT TURN ON TRACKER FAIL    |
|          | BZF    | NOLITE   |                                            |
|          | AD     | LRMCTR   | # M-S                                      |
|          | MASK   | NEG3     | # TEST FOR M-S > 3                         |
|          | EXTEND |          | # IF M-S > 3, THEN TWO OR MORE OF THE      |
|          | BZF    | +2       | # LAST FOUR V READINGS WERE BAD,           |
|          | TCF    | NOLITE   | # SO TURN ON VELOCITY FAIL LIGHT           |
|          | TC     | UPFLAG   | # AND SET BIT TO TURN ON TRACKER FAIL LITE |
|          | ADRES  | VFLSHFLG |                                            |



# SERVICER

PAGE 891

|    |        |       |          |                                           |    |
|----|--------|-------|----------|-------------------------------------------|----|
| 1  |        |       |          |                                           | 1  |
| 2  | NOLITE | CA    | LRMCTR   | # SET S = M                               | 2  |
| 3  |        | TS    | LRSCTR   |                                           | 3  |
| 4  |        |       |          |                                           | 4  |
| 5  |        | CCS   | VSELECT  | # TEST FOR Z COMPONENT                    | 5  |
| 6  |        | TCF   | ENDV DAT | # NOT Z, DO NOT SET VX INHIBIT            | 6  |
| 7  |        |       |          |                                           | 7  |
| 8  |        | TC    | UPFLAG   | # Z COMPONENT - SET FLAG TO SKIP X        | 8  |
| 9  |        | ADRES | VXINH    | # COMPONENT, AS ERROR MAY BE DUE TO CROSS | 9  |
| 10 |        | TCF   | ENDV DAT | # LOBE LOCK UP NOT DETECTED ON X AXIS.    | 10 |
| 11 |        |       |          |                                           | 11 |
| 12 |        |       |          |                                           | 12 |
| 13 |        |       |          |                                           | 13 |
| 14 |        |       |          |                                           | 14 |
| 15 |        |       |          |                                           | 15 |
| 16 |        |       |          |                                           | 16 |
| 17 |        |       |          |                                           | 17 |
| 18 |        |       |          |                                           | 18 |
| 19 |        |       |          |                                           | 19 |
| 20 |        |       |          |                                           | 20 |
| 21 |        |       |          |                                           | 21 |
| 22 |        |       |          |                                           | 22 |
| 23 |        |       |          |                                           | 23 |
| 24 |        |       |          |                                           | 24 |
| 25 |        |       |          |                                           | 25 |
| 26 |        |       |          |                                           | 26 |
| 27 |        |       |          |                                           | 27 |
| 28 |        |       |          |                                           | 28 |
| 29 |        |       |          |                                           | 29 |
| 30 |        |       |          |                                           | 30 |
| 31 |        |       |          |                                           | 31 |
| 32 |        |       |          |                                           | 32 |
| 33 |        |       |          |                                           | 33 |
| 34 |        |       |          |                                           | 34 |
| 35 |        |       |          |                                           | 35 |
| 36 |        |       |          |                                           | 36 |
| 37 |        |       |          |                                           | 37 |
| 38 |        |       |          |                                           | 38 |
| 39 |        |       |          |                                           | 39 |
| 40 |        |       |          |                                           | 40 |
| 41 |        |       |          |                                           | 41 |
| 42 |        |       |          |                                           | 42 |
| 43 |        |       |          |                                           | 43 |
| 44 |        |       |          |                                           | 44 |
| 45 |        |       |          |                                           | 45 |
| 46 |        |       |          |                                           | 46 |
| 47 |        |       |          |                                           | 47 |
| 48 |        |       |          |                                           | 48 |
| 49 |        |       |          |                                           | 49 |
| 50 |        |       |          |                                           | 50 |
| 51 |        |       |          |                                           | 51 |
| 52 |        |       |          |                                           | 52 |
| 53 |        |       |          |                                           | 53 |
| 54 |        |       |          |                                           | 54 |
| 55 |        |       |          |                                           | 55 |
| 56 |        |       |          |                                           | 56 |
| 57 |        |       |          |                                           | 57 |
| 58 |        |       |          |                                           | 58 |
| 59 |        |       |          |                                           | 59 |
| 60 |        |       |          |                                           | 60 |

1412THE

```
1 # *****
2 # LRVJOB IS SET WHEN THE LEM IS BELOW 15000 FT DURING THE LANDING PHASE
3 # THIS JOB INITIALIZES THE LANDING RADAR READ ROUTINE FOR 5 VELOCITY
4 # SAMPLES AND GOES TO SLEEP WHILE THE SAMPLING IS DONE -- ABOUT 500 MS.
5 # WITH A GOODEND RETURN THE DATA IS STORED IN VMEAS AND BIT7 OF LRSTAT
6 # IS SET. THE GIMBAL ANGLES ARE READ ABOUT MIDWAY IN THE SAMPLINGS.
```

```
7 170MS EQUALS ND1
```

```
8 LRVJOB CA 170MS # SET TASK TO READ CDUS + PIPAS
```

```
9 TC WAITLIST
```

```
10 EBANK= LRVTIME
```

```
11 2CADR RDGIMS
```

```
12 CCS VSELECT
```

```
13 TCF +2
```

```
14 CAF TWO
```

```
15 DOUBLE
```

```
16 TC BANKCALL
```

```
17 CADR LRVEL
```

```
18 TC BANKCALL
```

```
19 CADR RADSTALL
```

```
20 TCF VBAD
```

```
21 CCS STILBADV
```

```
22 TCF VSTILBAD
```

```
23 INHINT
```

```
24 EXTEND
```

```
25 DCA SAMPLSUM
```

```
26 DXCH VMEAS
```

```
27 CA EBANK4
```

```
28 TS EBANK
```

```
29 EBANK= LRVTIME
```

```
30 EXTEND
```

```
31 DCA LRVTIME
```

```
32 DXCH LRVTIMDL
```

```
33 EXTEND
```

```
34 DCA LRXCDU
```

```
35 DXCH LRXCDUDL
```

```
36 CA LRZCDU
```

```
37 TS LRZCDUDL
```

```
38 CA EBANK7
```

```
39 TS EBANK
```

```
40 EBANK= VSELECT
```

```
41 CS FLGWRD11
```

```
42 MASK VELDABIT
```

```
43 # SET BIT TO INDICATE VELOCITY
```

```
44 # MEASUREMENT MADE
```

```
1
2 ADS FLGWRD11
3 CCS VSELECT # UPDATE VSELECT
4 TCF +2
5 CA TWO
6 TS VSELECT
7 TCF ENDOFJOB
8
9 VBAD CAF TWO # SET STILBAD TO WAIT 4 SECONDS
10 VSTILBAD TS STILBADV
11 TCF ENDLRV
12
13 # LRHJOB IS SET BY LRHTASK WHEN LEM IS BELOW 25000 FT. THIS JOB
14 # INITIALIZES THE LR READ ROUTINE FOR AN ALT MEASUREMENT AND GOES TO
15 # SLEEP WHILE THE SAMPLING IS DONE -- ABOUT 95 MS. WITH A GOODEND RETURN
16 # THE ALT DATA IS STORED IN HMEAS AND BIT7 OF LRSTAT IS SET.
17
18 BANK 34
19 SETLOC R12STUFF
20 BANK
21
22 COUNT* $$/SERV
23
24 LRHJOB TC BANKCALL # INITIATE LR ALT MEASUREMENT
25 CADR LRALT
26 TC BANKCALL # LRHJOB TO SLEEP ABOUT 95MS
27 CADR RADSTALL
28 TCF HBAD
29 CCS STILBADH # IS DATA GOOD JUST PRESENT?
30 TCF HSTILBAD # JUST GOOD -- MUST WAIT 4 SECONDS.
31
32 INHINT
33 EXTEND
34 DCA SAMPLSUM # GOOD RETURN -- STORE AWAY LRH DATA
35 DXCH HMEAS # LRH DATA 1.079 FT/BIT
36 EXTEND # FOR DOWNLINK
37 DCA PIPTIME1
38 DXCH MKTIME
39
40 EXTEND
41 DCA CDUTEMPY # CDUY,Z = AIG,AMG
42 DXCH AIG
43
44 CA CDUTEMPX # CDUX = AOG
45 TS AOG
46
47 CS FLGWRD11 # SET BIT TO INDICATE RANGE
48 MASK RNGEDBIT # MEASUREMENT MADE.
49 ADS FLGWRD11
50 TC ENDOFJOB # TERMATE LRHJOB
51
52
53
54
55
56
57
58
59
60
```

HBAD

CA

FLAGWRD5

MASK

RNGSCBIT

# IS BAD RETURN DUE TO SCALE CHANGE?

EXTEND

BZF

HSTILBAD -1

# NO RESET HSTILBAD

TC

DOWNFLAG

# YES RESET SCALE CHANGE BIT AND IGNORE

ADRES

RNGSCFLG

TC

ENDOFJOB

HSTILBAD

CAF

TWO

# SET STILBAD TO WAIT 4 SECONDS

TS

STILBADH

TC

ENDOFJOB

BANK

34

SETLOC

SERV4

BANK

COUNT\* \$\$/SERV

# RDGIMS IS A TASK SET UP BY LRVJOB TO PICK UP THE IMU CDUS AND TIME  
# AT ABOUT THE MIDPOINT OF THE LR VEL READ ROUTINE WHEN 5 VEL SAMPLES  
# ARE SPECIFIED.

EBANK=

LRVTIME

RDGIMS

EXTEND

DCA

TIME2

# PICK UP TIME2, TIME1

DXCH

LRVTIME

# AND SAVE IN LRVTIME

EXTEND

DCA

CDUX

# PICK UP CDUX AND CDUY

DXCH

LRXCDU

# AND SAVE IN LRXCDU AND LRYCDU

CA

CDUZ

TS

LRZCDU

# SAVE CDUZ IN LRZCDU

CA

PIPAX

TS

PIPTM

# SAVE PIPAX IN PIPTM

EXTEND

DCA

PIPAY

# PICK UP PIPAY AND PIPAZ

DXCH

PIPTM +1

# AND SAVE IN PIPTM +1 AND PIPTM +2

TC

TASKOVER

BANK

33

SETLOC

SERVICES

BANK

COUNT\* \$\$/SERV

EBANK=

DVCNTR

```
1 # HIGATJOB IS SET APPROXIMATELY 6 SECONDS PRIOR TO HIGH GATE DURING
2 # THE DESCENT BURN PHASE OF LUNAR LANDING. THIS JOB INITIATES THE
3 # LANDING RADAR REPOSITIONING ROUTINE AND GOES TO SLEEP UNTIL THE
4 # LR ANTENNA MOVES FROM POSITION 1 TO POSITION 2. IF THE LR ANTENNA
5 # ACHIEVES POSITION 2 WITHIN 22 SECONDS THE ALTITUDE AND VELOCITY
6 # BEAM VECTORS ARE RECOMPUTED TO REFLECT THE NEW ORIENTATION WITH
7 # RESPECT TO THE NB. BIT10 OF LRSTAT IS CLEARED TO ALLOW LR
8 # MEASUREMENTS AND THE JOB TERMINATES.
9
10 HIGATJOB TC BANKCALL # START LRPOS2 JOB
11 CADR LRPOS2
12
13 TC BANKCALL # PUT HIGATJOB TO SLEEP UNTIL JOB IS DONE
14 CADR RADSTALL
15 TCF POSALARM # BAD END ALARM
16
17 POSGOOD CA PRI023 # REDUCE PRIORITY FOR INTERPRETIVE COMPS.
18 TC PRIOCHNG
19
20 TC SETPOS2 # LR IN POS2 -- SET UP TRANSFORMATIONS
21
22 TC DOWNFLAG
23 ADRES NOLRREAD # RESET NOLRREAD FLAG TO ENABLE LR READING
24 TC ENDOFJOB
25
26 POSALARM CA OCT523
27 TC BANKCALL
28 CADR PRIOLARM # FLASH ALARM CODE
29 TCF GOTOPOOH # TERMINATE
30 TCF +3 # PROCEED -- TRY AGAIN
31 TCF ENDOFJOB # V 32 E TERMINATE R12
32 TC ENDOFJOB
33
34 +3 CA BIT7 # SEE IF IN POS2 YET
35 EXTEND
36 RAND CHAN33
37 EXTEND
38 BZF POSGOOD # POS2 ACHIEVED SET UP ANTENNA BEAMS
39 TCF POSALARM # STILL DIDN'T MAKE IT REALARM
40
41 OCT523 OCT 00523
42
43
44 SETPOS1 TC MAKECADR # MUST BE CALLED BY BANKCALL
45 TS LRADRET1 # SAVE RETURN CADR. SINCE BUP2 CLOBBED
46
47 CAF TWO
48 TS STILBADH # INITIALIZE STILBAD
49 TS STILBADV # INITIALIZE STILBAD
50
51 CA ZERO # INDEX FOR LRALPHA, LRBETA IN POS 1.
52
53
54
55
56
57
58
59
60
```

```
1
2 TS LRLCTR # SET L,M,R, ANS S TO ZERO
3 TS LRMCTR
4 TS LRRCTR
5 TS LRSCTR
6 TS VSELECT # INITIALIZE VSELECT
7
8 TC SETPOS # CONTINUE WITH COMPUTATIONS.
9
10 CA LRADRET1
11 TC BANKJUMP # RETURN TO CALLER
12
13 SETPOS2 CA TWO # INDEX FOR POS2
14 SETPOS XCH Q # SAVE INDEX IN Q
15 TS LRADRET # SAVE RETURN
16
17 CA EBANK5
18 TS EBANK
19 EBANK= LRALPHA
20
21 EXTEND
22 INDEX Q
23 DCA LRALPHA # LRALPHA IN A, LRBETA IN L
24 TS CDUSPOT +4 # ROTATION ABOUT X
25 LXCH CDUSPOT # ROTATION ABOUT Y
26 CA ZERO
27 TS CDUSPOT +2 # ZERO ROTATION ABOUT Z.
28
29 CA EBANK7
30 TS EBANK
31 EBANK= LRADRET
32
33 TC INTPRET
34 VLOAD CALL
35 UNITY # CONVERT UNITY(ANTENNA) TO NB
36 TRG*SMNB
37 STOVL VYBEAMNB
38 UNITX # CONVERT UNITX(ANTENNA) TO NB
39 CALL
40 *SMNB*
41 STORE VXBEAMNB
42 VXV VSL1
43 STOVL VYBEAMNB
44 VZBEAMNB # Z = X * Y
45 HBEAMANT
46 CALL
47 *SMNB* # CONVERT TO NB
48 STORE HBEAMNB
49 EXIT
```

# SERVICER

|    |         |
|----|---------|
| TC | LRADRET |
|----|---------|



|         |        |          |                                            |
|---------|--------|----------|--------------------------------------------|
|         | BANK   | 21       |                                            |
|         | SETLOC | R10      |                                            |
|         | BANK   |          |                                            |
|         | EBANK= | UNIT/R/  |                                            |
|         | COUNT* | \$\$/R10 |                                            |
| LANDISP | LXCH   | PIPCTR1  | # UPDATE TBASE2 AND PIPCTR SIMULTANEOUSLY. |
|         | CS     | TIME1    |                                            |
|         | DXCH   | TBASE2   |                                            |
|         | CS     | FLAGWRD7 | # IS LANDING ANALOG DISPLAYS FLAG SET?     |
|         | MASK   | SWANDBIT |                                            |
|         | CCS    | A        |                                            |
|         | TCF    | DISPRSET | # NO.                                      |
|         | CA     | IMODES33 | # BIT 7 = 0 (DO ALTRATE), =1 (DO ALT.)     |
|         | MASK   | BIT7     |                                            |
| ALTROUT | CCS    | A        |                                            |
|         | TCF    | ALTOUT   |                                            |
|         | TC     | DISINDAT | # CHECK MODE SELECT SWITCH AND DIDFLG.     |
|         | CS     | IMODES33 |                                            |
|         | MASK   | BIT7     |                                            |
|         | ADS    | IMODES33 | # ALTERNATE ALTITUDE RATE WITH ALTITUDE.   |
|         | CAF    | BIT2     | # RATE COMMAND IS EXECUTED BEFORE RANGE.   |
|         | EXTEND |          |                                            |
|         | WOR    | CHAN14   | # ALTRATE (BIT2 = 1), ALTITUDE (BIT2 = 0). |
| ARCOMP  | CA     | RUNIT    | # COMPUTE ALTRATE=RUNIT.VVECT M/CS *2(-6). |
|         | EXTEND |          |                                            |
|         | MP     | VVECT    | # MULTIPLY X-COMPONENTS.                   |
|         | XCH    | RUPTREG1 | # SAVE SINGLE PRECISION RESULT M/CS*2(-6). |
|         | CA     | RUNIT +1 | # MULTIPLY Y-COMPONENTS.                   |
|         | EXTEND |          |                                            |
|         | MP     | VVECT +1 |                                            |
|         | ADS    | RUPTREG1 | # ACCUMULATE PARTIAL PRODUCTS.             |
|         | CA     | RUNIT +2 | # MULTIPLY Z-COMPONENTS.                   |
|         | EXTEND |          |                                            |
|         | MP     | VVECT +2 |                                            |
|         | ADS    | RUPTREG1 | # ALTITUDE RATE IN M/CS *2(-6).            |
|         | CA     | ARCONV   | # CONVERT ALTRATE TO BIT UNITS (.5FPS/BIT) |
|         | EXTEND |          |                                            |
|         | MP     | RUPTREG1 |                                            |
|         | DDOUBL |          |                                            |
|         | DDOUBL |          |                                            |
|         | XCH    | RUPTREG1 | # ALTITUDE RATE IN BIT UNITS*2(-14).       |
|         | CA     | DALTRATE | # ALTITUDE RATE COMPENSATION FACTOR.       |
|         | EXTEND |          |                                            |
|         | MP     | DT       |                                            |
|         | AD     | RUPTREG1 |                                            |
|         | TS     | ALTRATE  | # ALTITUDE RATE IN BIT UNITS*2(-14).       |
|         | CS     | ALTRATE  |                                            |

|    |         |        |            |                                            |
|----|---------|--------|------------|--------------------------------------------|
| 1  |         | EXTEND |            | # CHECK POLARITY OF ALTITUDE RATE.         |
| 2  |         | BZMF   | +2         |                                            |
| 3  |         | TCF    | DATAOUT    | # NEGATIVE - SEND POS. PULSES TO ALTM REG. |
| 4  |         | CA     | ALTRATE    | # POSITIVE OR ZERO - SET SIGN BIT = 1 AND  |
| 5  |         | AD     | BIT15      | # SEND TO ALTM REGISTER. *DO NOT SEND +0*  |
| 6  |         | TS     | ALTM       | # ACTIVATE THE LANDING ANALOG DISPLAYS - - |
| 7  | DATAOUT | CAF    | BIT3       |                                            |
| 8  |         | EXTEND |            |                                            |
| 9  |         | WOR    | CHAN14     | # BIT3 DRIVES THE ALT/ALTRATE METER.       |
| 10 |         | TCF    | TASKOVER   | # EXIT                                     |
| 11 |         |        |            |                                            |
| 12 |         |        |            |                                            |
| 13 | ALTOUT  | TC     | DISINDAT   | # CHECK MODE SELECT SWITCH AND DIDFLG.     |
| 14 |         | CS     | BIT7       |                                            |
| 15 |         | MASK   | IMODES33   |                                            |
| 16 |         | TS     | IMODES33   | # ALTERNATE ALTITUDE RATE WITH ALTITUDE.   |
| 17 |         | CS     | BIT2       |                                            |
| 18 |         | EXTEND |            |                                            |
| 19 |         | WAND   | CHAN14     |                                            |
| 20 |         | CCS    | ALTBITS    | # ==-1 IF OLD ALT. DATA TOBE EXTRAPOLATED. |
| 21 |         | TCF    | +4         |                                            |
| 22 |         | TCF    | +3         |                                            |
| 23 |         | TCF    | OLDDATA    |                                            |
| 24 |         | TS     | ALTBITS    | # SET ALTBITS FROM -0 TO +0.               |
| 25 |         | CS     | ONE        |                                            |
| 26 |         | DXCH   | ALTBITS    | # SET ALTBITS=-1 FOR SWITCH USE NEXT PASS. |
| 27 |         | DXCH   | ALTSAVE    |                                            |
| 28 |         | CA     | BIT10      | # NEW ALTITUDE EXTRAPOLATION WITH ALTRATE. |
| 29 |         | XCH    | Q          |                                            |
| 30 |         | LXCH   | 7          | # ZL                                       |
| 31 |         | CA     | DT         |                                            |
| 32 |         | EXTEND |            |                                            |
| 33 |         | DV     | Q          | # RESCALE DT*2(-14) TO *2(-9) TIME IN CS.  |
| 34 |         | EXTEND |            |                                            |
| 35 |         | MP     | ARTOA2     | # .0021322 *2(+8)                          |
| 36 |         | TCF    | OLDDATA +1 | # RATE APPLIES FOR DT CS.                  |
| 37 |         |        |            |                                            |
| 38 | ZDATA2  | DXCH   | ALTSAVE    |                                            |
| 39 |         | TCF    | NEWDATA    |                                            |
| 40 | OLDDATA | CA     | ARTOA      | # RATE APPLIES FOR .5 SEC. (4X/SEC. CYCLE) |
| 41 |         | EXTEND |            |                                            |
| 42 |         | MP     | ALTRATE    | # EXTRAPOLATE WITH ALTITUDE RATE.          |
| 43 |         | DDOUBL |            |                                            |
| 44 |         | AD     | ALTSAVE +1 |                                            |
| 45 |         | TS     | ALTSAVE +1 |                                            |
| 46 |         | CAF    | ZERO       |                                            |
| 47 |         | ADS    | ALTSAVE    |                                            |
| 48 |         | CAF    | POSMAX     | # FORCE SIGN AGREEMENT ASSUMING A          |
| 49 |         | AD     | ONE        | # NON-NEGATIVE ALTSAVE.                    |
| 50 |         | AD     | ALTSAVE +1 | # IF ALTSAVE IS NEGATIVE, ZERO ALTSAVE     |
| 51 |         | TS     | ALTSAVE +1 | # AND ALTSAVE +1 AT ZERODATA.              |
| 52 |         |        |            |                                            |
| 53 |         |        |            |                                            |
| 54 |         |        |            |                                            |
| 55 |         |        |            |                                            |
| 56 |         |        |            |                                            |
| 57 |         |        |            |                                            |
| 58 |         |        |            |                                            |
| 59 |         |        |            |                                            |
| 60 |         |        |            |                                            |

|    |          |        |            |                                            |
|----|----------|--------|------------|--------------------------------------------|
| 1  |          |        |            |                                            |
| 2  |          | CAF    | ZERO       |                                            |
| 3  |          | AD     | POSMAX     |                                            |
| 4  |          | AD     | ALTSAVE    |                                            |
| 5  |          | TS     | ALTSAVE    | # POSSIBLY SKIP TO NEWDATA.                |
| 6  |          | TCF    | ZERODATA   |                                            |
| 7  | NEWDATA  | CCS    | ALTSAVE +1 |                                            |
| 8  |          | TCF    | +4         |                                            |
| 9  |          | TCF    | +3         |                                            |
| 10 |          | CAF    | ZERO       | # SET NEGATIVE ALTSAVE +1 TO +0.           |
| 11 |          | TS     | ALTSAVE +1 |                                            |
| 12 |          | CCS    | ALTSAVE    | # PROVIDE A 15 BIT UNSIGNED OUTPUT.        |
| 13 |          | CAF    | BIT15      | # THE HI-ORDER PART IS +1 OR +0.           |
| 14 |          | AD     | ALTSAVE +1 |                                            |
| 15 |          | TCF    | DATAOUT    | # DISPATCH UNSIGNED BITS TO ALTM REG.      |
| 16 | DISINDAT | EXTEND |            |                                            |
| 17 |          | QXCH   | LADQSAVE   | # SAVE RETURN TO ALTROUT +1 OR ALTOUT +1   |
| 18 |          | CAF    | BIT6       |                                            |
| 19 |          | EXTEND |            | # WISHETH THE ASTRONAUT THE ANALOG         |
| 20 |          | RAND   | CHAN30     | # DISPLAYS? I.E.,                          |
| 21 |          | CCS    | A          | # IS THE MODE SELECT SWITCH IN PGNC?       |
| 22 |          | TCF    | DISPRSET   | # NO. ASTRONAUT REQUESTS NO INERTIAL DATA  |
| 23 |          | CS     | FLAGWRD1   | # YES. CHECK STATUS OF DIDFLAG.            |
| 24 |          | MASK   | DIDFLBIT   |                                            |
| 25 |          | EXTEND |            |                                            |
| 26 |          | BZF    | SPEEDRUN   | # SET. PERFORM DATA DISPLAY SEQUENCE.      |
| 27 |          | CS     | FLAGWRD1   | # RESET. PERFORM INITIALIZATION FUNCTIONS. |
| 28 |          | MASK   | DIDFLBIT   |                                            |
| 29 |          | ADS    | FLAGWRD1   | # SET DIDFLAG.                             |
| 30 |          | CS     | BIT7       |                                            |
| 31 |          | MASK   | IMODES33   | # TO DISPLAY ALTRATE FIRST AND ALT. SECOND |
| 32 |          | TS     | IMODES33   |                                            |
| 33 |          | CS     | FLAGWRD0   | # ARE WE IN DESCENT TRAJECTORY?            |
| 34 |          | MASK   | R10FLBIT   |                                            |
| 35 |          | EXTEND |            |                                            |
| 36 |          | BZF    | TASKOVER   | # NO                                       |
| 37 |          | CAF    | BIT8       | # YES.                                     |
| 38 |          | EXTEND |            |                                            |
| 39 |          | WOR    | CHAN12     | # SET DISPLAY INERTIAL DATA OUTBIT.        |
| 40 |          | CAF    | ZERO       |                                            |
| 41 |          | TS     | TRAKLATV   | # LATERAL VELOCITY MONITOR FLAG            |
| 42 |          | TS     | TRAKFWDV   | # FORWARD VELOCITY MONITOR FLAG            |
| 43 |          | TS     | LATVMETR   | # LATVEL MONITOR METER                     |
| 44 |          | TS     | FORVMETR   | # FORVEL MONITOR METER                     |
| 45 |          | CAF    | BIT4       |                                            |
| 46 |          | TC     | TWIDDLE    |                                            |
| 47 |          | ADRES  | INTLZE     |                                            |
| 48 |          | TCF    | TASKOVER   |                                            |
| 49 | INTLZE   | CAF    | BIT2       |                                            |
| 50 |          | EXTEND |            |                                            |
| 51 |          | WOR    | CHAN12     | # ENABLE RR ERROR COUNTER.                 |

1412THE

```
1
2 EXTEND
3 DCA GDT/2 +4 # COMPUTE THE Z-COMPONENT OF VELOCITY.
4
5 DDOUBL
6 DDOUBL
7 EXTEND
8
9 MP DT
10 EXTEND
11 DV ITEMP5
12
13 XCH VVECT +2
14 EXTEND
15 DCA V +4
16
17 DDOUBL
18 DDOUBL
19 ADS VVECT +2
20
21 CA PIPAZ
22 AD PIPATMPZ
23 EXTEND
24
25 MP KPIP1(5)
26 ADS VVECT +2
27
28
29 CAF BIT3 # PAUSE 40 MS TO LET OTHER RUPTS IN.
30 TC VARDELAY
31
32
33 CS FLAGWRDO # ARE WE IN DESCENT TRAJECTORY?
34 MASK R10FLBIT
35 CCS A
36
37 TCF +2 # YES.
38 TC LADQSAVE # NO.
39
40
41 CA DELVS # HI X OF VELOCITY CORRECTION TERM.
42 AD VVECT # HI X OF UPDATED VELOCITY VECTOR.
43 TS ITEMP1 # = VX - DVX M/CS*2(-5).
44
45 CA DELVS +2 # Y
46 AD VVECT +1 # Y
47 TS ITEMP2 # = VY - DVY M/CS*2(-5).
48
49 CA DELVS +4 # Z
50 AD VVECT +2 # Z
51 TS ITEMP3 # = VZ - DVZ M/CS*2(-5).
52
53 CA ITEMP1 # COMPUTE VHY, VELOCITY DIRECTED ALONG THE
54 EXTEND # Y-COORDINATE.
55 MP UHYP # HI X OF CROSS-RANGE HALF-UNIT VECTOR.
56
57 XCH RUPTREG1
58 CA ITEMP2
59 EXTEND
60
61 MP UHYP +2 # Y
62 ADS RUPTREG1 # ACCUMULATE PARTIAL PRODUCTS.
63 CA ITEMP3
64
65 EXTEND
66 MP UHYP +4 # Z
67 ADS RUPTREG1
68
69
70
71
72
73
74
75
76
77
78
79
80
```

```
1
2 CA RUPTREG1
3 DOUBLE
4 XCH VHY # VHY=VMP.UHYP M/CS*2(-5).
5 CA ITEMP1 # NO COMPUTE VHZ, VELOCITY DIRECTED ALONG
6 EXTEND # THE Z-COORDINATE.
7 MP UHYP # HI X OF DOWN-RANGE HALF-UNIT VECTOR.
8 XCH RUPTREG1
9 CA ITEMP2
10 EXTEND
11 MP UHYP +2 # Y
12 ADS RUPTREG1 # ACCUMULATE PARTIAL PRODUCTS.
13 CA ITEMP3
14 EXTEND
15 MP UHYP +4 # Z
16 ADS RUPTREG1
17 CA RUPTREG1
18 DOUBLE
19 GET22/32 XCH VHZ # VHZ = VMP.UHYP M/CS*2(-5).
20 CAF EBANK6 # GET SIN(AOG),COS(AOG) FROM GPMATRIX.
21 TS EBANK
22 EBANK= M22
23 CA M22
24 TS ITEMP3
25 CA M32
26 TS ITEMP4
27 CAF EBANK7
28 TS EBANK
29 LATFWDV EBANK= UNIT/R/
30 CA ITEMP4 # COMPUTE LATERAL AND FORWARD VELOCITIES.
31 EXTEND
32 MP VHY
33 XCH RUPTREG1
34 CA ITEMP3
35 EXTEND
36 MP VHZ
37 ADS RUPTREG1 # =VHY(COS)AOG+VHZ(SIN)AOG M/CS *2(-5)
38 CA VELCONV # CONVERT LATERAL VELOCITY TO BIT UNITS.
39 EXTEND
40 MP RUPTREG1
41 DDOUBL
42 XCH LATVEL # LATERAL VELOCITY IN BIT UNITS *2(-14).
43 CA ITEMP4 # COMPUTE FORWARD VELOCITY.
44 EXTEND
45 MP VHZ
46 XCH RUPTREG1
47 CA ITEMP3
48 EXTEND
49 MP VHY
50 CS A
51 ADS RUPTREG1 # =VHZ(COS)AOG-VHY(SIN)AOG M/CS *2(-5).
```

|          |        |             |                                            |
|----------|--------|-------------|--------------------------------------------|
|          | CA     | VELCONV     | # CONVERT FORWARD VELOCITY TO BIT UNITS.   |
|          | EXTEND |             |                                            |
|          | MP     | RUPTREG1    |                                            |
|          | DDOUBL |             |                                            |
|          | XCH    | FORVEL      | # FORWARD VELOCITY IN BIT UNITS *2(-14).   |
|          | CS     | MAXVBITS    | # ACC.=-199.9989 FT./SEC.                  |
|          | TS     | ITEMP6      | # -547 BIT UNITS (OCTAL) AT 0.5571 FPS/BIT |
|          | CAF    | ONE         | # LOOP TWICE.                              |
| VMONITOR | TS     | ITEMP5      | # FORWARD AND LATERAL VELOCITY LANDING     |
|          | INDEX  | ITEMP5      | # ANALOG DISPLAYS MONITOR.                 |
|          | CCS    | LATVEL      |                                            |
|          | TCF    | +4          |                                            |
|          | TCF    | LVLIMITS    |                                            |
|          | TCF    | +8D         |                                            |
|          | TCF    | LVLIMITS    |                                            |
|          | INDEX  | ITEMP5      |                                            |
|          | CS     | LATVEL      |                                            |
|          | AD     | MAXVBITS    | # +199.9989 FT.SEC.                        |
|          | EXTEND |             |                                            |
|          | BZMF   | CHKLASTY    |                                            |
|          | TCF    | LVLIMITS    |                                            |
|          | INDEX  | ITEMP5      |                                            |
|          | CA     | LATVEL      |                                            |
|          | AD     | MAXVBITS    |                                            |
|          | EXTEND |             |                                            |
|          | BZMF   | +2          |                                            |
|          | TCF    | LVLIMITS    |                                            |
| CHKLASTY | INDEX  | ITEMP5      |                                            |
|          | CCS    | LATVMETR    |                                            |
|          | TCF    | +4          |                                            |
|          | TCF    | LASTOK      |                                            |
|          | TCF    | +7          |                                            |
|          | TCF    | LASTOK      |                                            |
|          | INDEX  | ITEMP5      |                                            |
|          | CA     | LATVEL      |                                            |
|          | EXTEND |             |                                            |
|          | BZMF   | LASTPOSY +5 |                                            |
|          | TCF    | +5          |                                            |
|          | INDEX  | ITEMP5      |                                            |
|          | CS     | LATVEL      |                                            |
|          | EXTEND |             |                                            |
|          | BZMF   | LASTNEGY +4 |                                            |
| LASTOK   | INDEX  | ITEMP5      |                                            |
|          | CCS    | TRAKLATV    |                                            |
|          | TCF    | LASTPOSY    |                                            |
|          | TCF    | +2          |                                            |
|          | TCF    | LASTNEGY    |                                            |
|          | INDEX  | ITEMP5      |                                            |

|    |          |        |             |    |
|----|----------|--------|-------------|----|
| 1  |          |        |             | 1  |
| 2  |          | CA     | LATVEL      | 2  |
| 3  |          | EXTEND |             | 3  |
| 4  |          | BZMF   | NEGVMAXY    | 4  |
| 5  |          | TCF    | POSVMAXY    | 5  |
| 6  | LASTPOSY | INDEX  | ITEMP5      | 6  |
| 7  |          | CA     | LATVEL      | 7  |
| 8  |          | EXTEND |             | 8  |
| 9  |          | BZMF   | +2          | 9  |
| 10 |          | TCF    | POSVMAXY    | 10 |
| 11 |          | CS     | MAXVBITS    | 11 |
| 12 |          | TCF    | ZEROLSTY    | 12 |
| 13 | POSVMAXY | INDEX  | ITEMP5      | 13 |
| 14 |          | CS     | LATVMETR    | 14 |
| 15 |          | AD     | MAXVBITS    | 15 |
| 16 |          | INDEX  | ITEMP5      | 16 |
| 17 |          | XCH    | RUPTREG3    | 17 |
| 18 |          | CAF    | ONE         | 18 |
| 19 |          | TCF    | ZEROLSTY +3 | 19 |
| 20 | LASTNEGY | INDEX  | ITEMP5      | 20 |
| 21 |          | CA     | LATVEL      | 21 |
| 22 |          | EXTEND |             | 22 |
| 23 |          | BZMF   | NEGVMAXY    | 23 |
| 24 |          | CA     | MAXVBITS    | 24 |
| 25 |          | TCF    | ZEROLSTY    | 25 |
| 26 | NEGVMAXY | INDEX  | ITEMP5      | 26 |
| 27 |          | CA     | LATVMETR    | 27 |
| 28 |          | AD     | MAXVBITS    | 28 |
| 29 |          | COM    |             | 29 |
| 30 |          | INDEX  | ITEMP5      | 30 |
| 31 |          | XCH    | RUPTREG3    | 31 |
| 32 |          | CS     | ONE         | 32 |
| 33 |          | TCF    | ZEROLSTY +3 | 33 |
| 34 | LVLIMITS | INDEX  | ITEMP5      | 34 |
| 35 |          | CCS    | TRAKLATV    | 35 |
| 36 |          | TCF    | LATVPOS     | 36 |
| 37 |          | TCF    | +2          | 37 |
| 38 |          | TCF    | LATVNEG     | 38 |
| 39 |          | INDEX  | ITEMP5      | 39 |
| 40 |          | CS     | LATVMETR    | 40 |
| 41 |          | EXTEND |             | 41 |
| 42 |          | BZMF   | +2          | 42 |
| 43 |          | TCF    | NEGLMLV     | 43 |
| 44 |          | INDEX  | ITEMP5      | 44 |
| 45 |          | CS     | LATVEL      | 45 |
| 46 |          | EXTEND |             | 46 |
| 47 |          | BZMF   | LVMINLM     | 47 |
| 48 |          | AD     | ITEMP6      | 48 |
| 49 |          | INDEX  | ITEMP5      | 49 |
| 50 |          | AD     | LATVMETR    | 50 |
| 51 |          | EXTEND |             | 51 |
| 52 |          |        |             | 52 |
| 53 |          |        |             | 53 |
| 54 |          |        |             | 54 |
| 55 |          |        |             | 55 |
| 56 |          |        |             | 56 |
| 57 |          |        |             | 57 |
| 58 |          |        |             | 58 |
| 59 |          |        |             | 59 |
| 60 |          |        |             | 60 |





|    |          |        |                                       |    |
|----|----------|--------|---------------------------------------|----|
| 1  |          |        |                                       | 1  |
| 2  |          | BZMF   | LVMINLM                               | 2  |
| 3  |          | INDEX  | ITEMP5                                | 3  |
| 4  |          | AD     | LATVEL                                | 4  |
| 5  |          | EXTEND |                                       | 5  |
| 6  |          | INDEX  | ITEMP5                                | 6  |
| 7  |          | SU     | LATVMETR                              | 7  |
| 8  |          | TCF    | ZEROLSTY                              | 8  |
| 9  | LATVPOS  | INDEX  | ITEMP5                                | 9  |
| 10 |          | CS     | LATVEL                                | 10 |
| 11 |          | EXTEND |                                       | 11 |
| 12 |          | BZMF   | LVMINLM                               | 12 |
| 13 |          | TCF    | +5                                    | 13 |
| 14 | LATVNEG  | INDEX  | ITEMP5                                | 14 |
| 15 |          | CA     | LATVEL                                | 15 |
| 16 |          | EXTEND |                                       | 16 |
| 17 |          | BZMF   | LVMINLM                               | 17 |
| 18 |          | INDEX  | ITEMP5                                | 18 |
| 19 |          | CS     | LATVMETR                              | 19 |
| 20 |          | TCF    | ZEROLSTY                              | 20 |
| 21 | NEGLMLV  | INDEX  | ITEMP5                                | 21 |
| 22 |          | CA     | LATVEL                                | 22 |
| 23 |          | EXTEND |                                       | 23 |
| 24 |          | BZMF   | LVMINLM                               | 24 |
| 25 |          | CA     | MAXVBITS                              | 25 |
| 26 |          | INDEX  | ITEMP5                                | 26 |
| 27 |          | AD     | LATVMETR                              | 27 |
| 28 |          | COM    |                                       | 28 |
| 29 |          | INDEX  | ITEMP5                                | 29 |
| 30 |          | AD     | LATVEL                                | 30 |
| 31 |          | EXTEND |                                       | 31 |
| 32 |          | BZMF   | LVMINLM                               | 32 |
| 33 |          | EXTEND |                                       | 33 |
| 34 |          | INDEX  | ITEMP5                                | 34 |
| 35 |          | SU     | LATVEL                                | 35 |
| 36 |          | INDEX  | ITEMP5                                | 36 |
| 37 |          | AD     | LATVMETR                              | 37 |
| 38 |          | COM    |                                       | 38 |
| 39 |          | TCF    | ZEROLSTY                              | 39 |
| 40 | LVMINLM  | INDEX  | ITEMP5                                | 40 |
| 41 |          | CS     | LATVMETR                              | 41 |
| 42 |          | INDEX  | ITEMP5                                | 42 |
| 43 |          | AD     | LATVEL                                | 43 |
| 44 | ZEROLSTY | INDEX  | ITEMP5                                | 44 |
| 45 |          | XCH    | RUPTREG3                              | 45 |
| 46 |          | CAF    | ZERO                                  | 46 |
| 47 |          | INDEX  | ITEMP5                                | 47 |
| 48 |          | TS     | TRAKLATV                              | 48 |
| 49 |          | INDEX  | ITEMP5                                | 49 |
| 50 |          | CA     | RUPTREG3                              | 50 |
| 51 |          | AD     | NEG0                                  | 51 |
| 52 |          |        | # AVOIDS +0 DINC HARDWARE MALFUNCTION | 52 |
| 53 |          |        |                                       | 53 |
| 54 |          |        |                                       | 54 |
| 55 |          |        |                                       | 55 |
| 56 |          |        |                                       | 56 |
| 57 |          |        |                                       | 57 |
| 58 |          |        |                                       | 58 |
| 59 |          |        |                                       | 59 |
| 60 |          |        |                                       | 60 |

```

1
2 INDEX ITEMP5
3 TS CDUTCMD
4
5 INDEX ITEMP5
6 CA RUPTREG3
7 INDEX ITEMP5
8
9 ADS LATVMETR
10 CCS ITEMP5 # FIRST MONITOR FORWARD THEN LATERAL VEL.
11 TCF VMONITOR
12
13
14 CAF BITSET # DRIVE THE X-POINTER DISPLAY.
15 EXTEND
16
17 WOR CHAN14
18 TC LADQSAVE # GO TO ALTROUT +1 OR TO ALTOUT +1
19 CAF ZERO # ZERO ALTSAVE AND ALTSAVE +1 - - -
20 TS L #
21 TCF ZDATA2
22
23
24
25 # *****
26
27 DISPRSET CS FLAGWRD0 # ARE WE IN DESCENT TRAJECTORY?
28
29 MASK RIOFLBIT
30 EXTEND
31 BZF ABORTON # NO.
32
33 CAF BIT8 # YES.
34 MASK IMODES33 # CHECK IF INERTIAL DATA JUST DISPLAYED.
35 CCS A
36
37 CAF BIT2 # YES. DISABLE RR ERROR COUNTER
38 AD BIT8 # NO. REMOVE DISPLAY INERTIAL DATA
39 COM
40
41 EXTEND
42
43 WAND CHAN12
44 CS BITS8/7 # RESET INERTIAL DATA, INTERLEAVE FLAGS.
45
46 MASK IMODES33
47 TS IMODES33
48 CS DIDFLBIT
49
50 MASK FLAGWRD1
51 TS FLAGWRD1 # RESET DIDFLAG.
52 TCF TASKOVER
53
54 # *****
55 BITS8/7 OCT 00300 # INERTIAL DATA AND INTERLEAVE FLAGS.
56
57 BITSET = PRI06
58 # *****
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

```

```
1 # PROGRAM NAME: FINDCDUW
2 # MOD NUMBER: 1 68-07-15
3 # MOD AUTHOR: KLUMPP
4 #
5 # OBJECTS OF MOD: 1. TO SUPPLY COMMANDED GIMBAL ANGLES FOR NOUN 22.
6 # 2. TO MAINTAIN CORRECT AND CURRENT THRUST
7 # DIRECTION DATA IN ALL MODES. THIS IS DONE BY
8 # FETCHING FOR THE THRUST DIRECTION FILTER THE
9 # CDUD'S IN PNGCS-AUTO, THE CDU'S IN ALL OTHER
10 # MODES.
11 # 3. TO SUBSTITUTE A STOPRATE FOR THE NORMAL
12 # AUTOPILOT COMMANDS WHENEVER
13 # 1) NOT IN PNGCS-AUTO, OR
14 # 2) ENGINE IS OFF.
15 #
16 # FUNCTIONAL DESCRIPTION:
17 #
18 # FINDCDUW PROVIDES THE INTERFACES BETWEEN THE VARIOUS POWERED FLITE GUIDANCE PROGRAMS
19 # AND THE DIGITAL AUTOPILOT. THE INPUTS TO FINDCDUW ARE THE THRUST COMMAND VECTOR
20 # AND THE WINDOW COMMAND VECTOR, AND THE OUTPUTS ARE THE GIMBAL ANGLE
21 # INCREMENTS, THE COMMANDED ATTITUDE ANGLE RATES, AND THE COMMANDED
22 # ATTITUDE LAG ANGLES (WHICH ACCOUNT FOR THE ANGLES BY WHICH THE BODY WILL
23 # LAG BEHIND A RAMP COMMAND IN ATTITUDE ANGLE DUE TO THE FINITE ANGULAR
24 # ACCELERATIONS AVAILABLE).
25 #
26 # FINDCDUW ALIGNS THE ESTIMATED THRUST VECTOR FROM THE THRUST DIRECTION
27 # FILTER WITH THE THRUST COMMAND VECTOR, AND, WHEN XDVINHIB SET,
28 # ALIGNS THE +Z HALF OF THE LM ZX PLANE WITH THE WINDOW COMMAND VECTOR.
29 #
30 #
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

## # SPECIFICATIONS:

# INITIALIZATION: A SINGLE INTERPRETIVE CALL TO INITCDUW IS REQUIRED  
BEFORE EACH GUIDED MANEUVER USING FINDCDUW.

# CALL: INTERPRETIVE CALL TO FINDCDUW WITH THE THRUST COMMAND  
VECTOR IN MPAC. INTERPRETIVE CALL TO FINDCDUW -2 WITH  
THE THRUST COMMAND VECTOR IN UNFC/2 AND NOT IN MPAC.

# RETURNS: NORMAL INTERPRETIVE IN ALL CASES  
1. NORMALLY ALL AUTOPILOT CMDS ARE ISSUED.  
2. IF NOT PNGCS AUTO, DO STOPRATE AND RETURN  
WITHOUT ISSUING AUTOPILOT CMDS.  
3. IF ENGINE OFF, DO STOPRATE AND RETURN WITHOUT  
ISSUING AUTOPILOT CMDS.

# ALARMS: 00401 IF INPUTS DETERMINE AN ATTITUDE IN GIMBAL LOCK.  
FINDCDUW DRIVES CDUXD AND CDUYD TO THE RQD VALUES,  
BUT DRIVES CDUZD ONLY TO THE GIMBAL LOCK CONE.

00402 IF UNFC/2 OR UNWC/2 PRODUCE OVERFLOW WHEN  
UNITIZED USING NORMUNIT. FINDCDUW ISSUES  
STOPRATE AS ONLY INPUT TO AUTOPILOT.

# INPUTS: UNFC/2 THRUST COMMAND VECTOR, NEED NOT BE SEMI-UNIT.  
UNWC/2 WINDOW COMMAND VECTOR, NEED NOT BE SEMI-UNIT.  
OGABIAS POSSIBLE BIAS FOR OUTER GIMBAL ANGLE (ZEROED IN INITCDUW), UNITS OF PI.  
XOVINHIB FLAG DENOTING X AXIS OVERRIDE INHIBITED.  
CSMDOCKD FLAG DENOTING CSM DOCKED.  
STEERSW FLAG DENOTING INSUFF THRUST FOR THRUST DIR FLTR.

# OUTPUTS: DELCDUX,Y,Z  
OMEGAPD,+1,+2  
DELPEROR,+1,+2  
CPHI,+1,+2 FOR NOUN22

# DEBRIS: FINDCDUW DESTROYS SINCDUX,Y,Z AND COSCDUX,Y,Z BY  
WRITING INTO THESE LOCATIONS THE SINES AND COSINES  
OF THE CDUD'S IN PNGCS-AUTO, OF THE CDU'S OTHERWISE.

# INITIALIZATION FOR FINDCDUW

BANK 30  
SETLOC FCDUW  
BANK

EBANK= ECDUW  
COUNT\* \$\$/FCDUW

INITCDUW VLOAD  
UNITX  
STORE UNFV/2  
STORE UNWC/2  
RVQ

# FINDCDUW PRELIMINARIES

VLOAD # FINDCDUW -2: ENTRY WHEN UNFC/2 PRE-STORD  
# INPUT VECTORS NEED NOT BE SEMI-UNIT  
FINDCDUW BOV UNFC/2  
SETPD # FINDCDUW: ENTRY WHEN UNFC/2 IN MPAC  
FINDCDUW # INTERPRETER NOW INITIALIZED  
22 # LOCS 0 THRU 21 FOR DIRECTION COSINE MAT  
STQ EXIT  
QCDUWUSR # SAVE RETURN ADDRESS

# MORE HAUSKEEPING

CA ECDUWL  
XCH EBANK # SET EBANK  
TS ECDUWUSR # SAVE USER'S EBANK

CA DAPBOOLS  
MASK CSMDOCKD # CSMDOCKD MUST NOT BE BIT15

CCS A  
CA ONE # INDEX IF CSM DOCKED  
TS NDXCDUW

CA XOVINHIB # XOVINHIB MUST NOT BE BIT15  
TS FLPAUTNO # SET TO POS-NON-ZERO FLAG PNGCS AUTO NOT

MASK DAPBOOLS  
TS FLAGOODW # FLAGOODW = ANY PNZ NUMBER IF XOV INHIBTD

# FETCH BASIC DATA

INHINT

# RELINT AT PAUTNO (TC INTPRET)

CA CDUX  
TS CDUSPOTX  
CA CDUY  
TS CDUSPOTY  
CA CDUZ  
TS CDUSPOTZ

# FETCH CDUX,CDUY,CDUZ IN ALL CASES, BUT  
# REPLACE BELOW IF PNGCS AUTO

CA BIT10

# PNGCS CONTROL BIT

EXTEND  
RAND CHAN30  
CCS A

TCF PAUTNO

# NOT PNGCS (BITS INVERTED)

CA BIT14

# AUTO MODE BIT

EXTEND  
RAND CHAN31  
CCS A

TCF PAUTNO

# NOT AUTO (BITS INVERTED)

TS FLPAUTNO

# RESET FLAG PNGCS AUTO NOT

CA CDUXD  
TS CDUSPOTX

# PNGCS AUTO: FETCH CDUXD,CDUYD,CDUZD

CA CDUYD  
TS CDUSPOTY

CA CDUZD

TS CDUSPOTZ

# FETCH INPUTS

|        |       |                           |                                                                                    |
|--------|-------|---------------------------|------------------------------------------------------------------------------------|
| PAUTNO | TC    | INTPRET                   | # ENTERING THRUST CMD STILL IN MPAC                                                |
|        | RTB   |                           |                                                                                    |
|        | STOVL | NORMUNIT<br>UNX/2         | # SEMI-UNIT THRUST CMD AS INITIAL UNX/2                                            |
|        | RTB   | UNWC/2<br>RTB<br>NORMUNIT |                                                                                    |
|        | STOVL | QUICTRIG<br>UNZ/2<br>DELV | # ALWAYS RQD TO OBTAIN TRIGS OF CDUD'S<br># SEMI-UNIT WINDOW CMD AS INITIAL UNZ/2  |
|        | BOVB  | UNIT                      |                                                                                    |
|        | BOV   | NOATTCNT<br>CALL          | # AT LEAST ONE ENTERING CMD VCT ZERO                                               |
|        |       | AFTNFLTR<br>*SMNB*        | # IF UNIT DELV OVERFLOWS SKIP FILTER<br># YIELDS UNIT(DELV) IN VEH COORDS FOR FLTR |

# THRUST DIRECTION FILTER

EXIT

|      |         |                                            |
|------|---------|--------------------------------------------|
| CA   | UNFVY/2 | # FOR RESTARTS, UNFV/2 ALWAYS INTACT, MPAC |
| LXCH | MPAC +3 | # RENEWD AFTER RETURN FROM CALLER,         |
| TC   | FLTRSUB | # TWO FILTER UPDATES MAY BE DONE.          |
| TS   | UNFVY/2 | # UNFV/2 NEED NOT BE EXACTLY SEMI-UNIT.    |
| CA   | UNFVZ/2 |                                            |
| LXCH | MPAC +5 |                                            |
| TC   | FLTRSUB |                                            |
| TS   | UNFVZ/2 |                                            |
| TC   | INTPRET | # COMPLETES FILTER                         |

# FIND A SUITABLE WINDOW POINTING VECTOR

AFTRFLTR SLOAD BHIZ # IF XOY NOT INHIBITED, GO FETCH ZNB

FLAGOODW  
FETCHZNB

VLOAD CALL  
UNZ/2  
UNWCTEST

FETCHZNB VLOAD

ZNBPIP  
UNZ/2  
UNWCTEST

VLOAD VCOMP # Z AND -X CAN'T BOTH PARALLEL UNFC/2  
XNBPIP  
STORE UNZ/2

# COMPUTE THE REQUIRED DIRECTION COSINE MATRIX

DCMCL VLOAD VXV  
UNZ/2  
UNX/2

UNIT PUSH # UNY/2 FIRST ITERATION  
VXV VSL1  
UNX/2

STORE UNZ/2 # -UNZ/2 FIRST ITERATION  
VXSC PDVL # EXCHANGE -UNFVZ/2 UNZ/2 FOR UNY/2  
UNFVZ/2 # MUST BE SMALL

VXSC BVSU # YIELDS -UNFVY/2 UNY/2-UNFVZ/2 UNZ/2  
UNFVY/2 # MUST BE SMALL  
VSL1 VAD

UNX/2

UNIT # TOTALLY ELIMINATES THRUST POINTING ERROR  
STORE UNX/2 # UNX/2

VXV VSL1  
UNZ/2 # -UNZ/2 WAS STORED HERE REMEMBER  
STORE UNY/2 # UNY/2

VCOMP VXV  
UNX/2

VSL1

STORE UNZ/2 # UNZ/2



# COMPUTES THE REQUIRED GIMBAL ANGLES

```
CALL NB2CDUSP # YIELDS THE RQD GIMBAL ANGLES, 2'S, PI
EXIT
```

# LIMIT THE MIDDLE GIMBAL ANGLE &amp; COMPUTE THE UNLIMITED GIMBAL ANGLE CHGS

```
CA MPAC +2 # LIMIT THE MGA
TS L # CAN'T LXCH: NEED UNLIMITED MGA FOR ALARM
CA CDUZDLIM
TC LIMITSUB # YIELDS LIMITED MGA. 1 BIT ERROR POSSIBLE
XCH MPAC +2 # BECAUSE USING 2'S COMP. WHO CARES?
EXTEND
MSU MPAC +2 # THIS BETTER YIELD ZERO
EXTEND
BZF +2
TCF ALARMMGA
```

MGARET INHINT # RELINT AT TC INTPRET AFTER TCQCDUW

```
DELGMBLP ZL
 CA TWO
 TS TEM2
 CA L # TO PREVENT FALSE STARTS ABOUT X, ZERO
 EXTEND # FLAGOODW IF DELGMBZ OR Y TOO BIG.
 SQUARE
 AD HI5 # WITHIN 1 BIT OF -(45 DEG SQUARED)
 EXTEND
 BZMF +3
 CA ZERO
 TS FLAGOODW
```

```
INDEX TEM2
CA MPAC
INDEX TEM2
TS CPHI # OUTPUTS TO NOUN22
```

```
EXTEND
INDEX TEM2
MSU CDUXD # NO MATTER THAT THESE SLIGHTLY DIFFERENT
COM # FROM WHEN WE INITIALLY FETCHED THEM
```

```
INDEX TEM2
TS -DELGMB # -UNLIMITED GIMBAL ANGLE CHGS, 1'S, PI
TS L # FOR PRECEDING TEST ON NEXT LOOP PASS
CCS TEM2
TCF DELGMBLP
```



|    |                         |        |              |                 |    |
|----|-------------------------|--------|--------------|-----------------|----|
| 1  |                         |        |              |                 | 1  |
| 2  | # BRANCHES TO NOATT CNT |        |              |                 | 2  |
| 3  |                         | CCS    | FLPAUTNO     |                 | 3  |
| 4  |                         | TCF    | NOATT CNT +2 | # NO PNGCS AUTO | 4  |
| 5  |                         |        |              |                 | 5  |
| 6  |                         | CA     | FLAGWRD5     |                 | 6  |
| 7  |                         | MASK   | ENGONBIT     |                 | 7  |
| 8  |                         | EXTEND |              |                 | 8  |
| 9  |                         | BZF    | NOATT CNT +2 | # ENGINE NOT ON | 9  |
| 10 |                         |        |              |                 | 10 |
| 11 |                         |        |              |                 | 11 |
| 12 |                         |        |              |                 | 12 |
| 13 |                         |        |              |                 | 13 |
| 14 |                         |        |              |                 | 14 |
| 15 |                         |        |              |                 | 15 |
| 16 |                         |        |              |                 | 16 |
| 17 |                         |        |              |                 | 17 |
| 18 |                         |        |              |                 | 18 |
| 19 |                         |        |              |                 | 19 |
| 20 |                         |        |              |                 | 20 |
| 21 |                         |        |              |                 | 21 |
| 22 |                         |        |              |                 | 22 |
| 23 |                         |        |              |                 | 23 |
| 24 |                         |        |              |                 | 24 |
| 25 |                         |        |              |                 | 25 |
| 26 |                         |        |              |                 | 26 |
| 27 |                         |        |              |                 | 27 |
| 28 |                         |        |              |                 | 28 |
| 29 |                         |        |              |                 | 29 |
| 30 |                         |        |              |                 | 30 |
| 31 |                         |        |              |                 | 31 |
| 32 |                         |        |              |                 | 32 |
| 33 |                         |        |              |                 | 33 |
| 34 |                         |        |              |                 | 34 |
| 35 |                         |        |              |                 | 35 |
| 36 |                         |        |              |                 | 36 |
| 37 |                         |        |              |                 | 37 |
| 38 |                         |        |              |                 | 38 |
| 39 |                         |        |              |                 | 39 |
| 40 |                         |        |              |                 | 40 |
| 41 |                         |        |              |                 | 41 |
| 42 |                         |        |              |                 | 42 |
| 43 |                         |        |              |                 | 43 |
| 44 |                         |        |              |                 | 44 |
| 45 |                         |        |              |                 | 45 |
| 46 |                         |        |              |                 | 46 |
| 47 |                         |        |              |                 | 47 |
| 48 |                         |        |              |                 | 48 |
| 49 |                         |        |              |                 | 49 |
| 50 |                         |        |              |                 | 50 |
| 51 |                         |        |              |                 | 51 |
| 52 |                         |        |              |                 | 52 |
| 53 |                         |        |              |                 | 53 |
| 54 |                         |        |              |                 | 54 |
| 55 |                         |        |              |                 | 55 |
| 56 |                         |        |              |                 | 56 |
| 57 |                         |        |              |                 | 57 |
| 58 |                         |        |              |                 | 58 |
| 59 |                         |        |              |                 | 59 |
| 60 |                         |        |              |                 | 60 |

# LIMIT THE ATTITUDE ANGLE CHANGES

#

# THIS SECTION LIMITS THE ATTITUDE ANGLE CHANGES ABOUT A SET OF ORTHOGONAL VEHICLE AXES X,YPRIME,ZPRIME,  
# THESE AXES COINCIDE WITH THE COMMANDED VEHICLE AXES IF AND ONLY IF CDUXD IS ZERO. THE PRIME SYSTEM IS  
# THE COMMANDED VEHICLE SYSTEM ROTATED ABOUT THE X AXIS TO BRING THE Z AXIS INTO ALIGNMENT WITH THE MIDDLE GIMBAL  
# AXIS. ATTITUDE ANGLE CHANGES IN THE PRIME SYSTEM ARE RELATED TO SMALL GIMBAL ANGLE CHANGES BY:

$$\begin{bmatrix} -\text{DELATTX} \\ -\text{DELATTYPRIME} \\ -\text{DELATTZPRIME} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\text{CDUZD}) & 0 \\ 0 & \cos(\text{CDUZD}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\text{DELGMBX} \\ -\text{DELGMBY} \\ -\text{DELGMBZ} \end{bmatrix}$$

LXCH -DELGMB +2 # SAME AS -DELATTZPRIME UNLIMITED

INDEX NDXCDUW

CA DAZMAX

TC LIMITSUB

TS -DELGMB +2 # -DELGMBZ

CA -DELGMB +1

EXTEND

MP COSCDUZ

TS L

# YIELDS -DELATTYPRIME/2 UNLIMITED

INDEX NDXCDUW

CA DAY/2MAX

TC LIMITSUB

EXTEND

DV COSCDUZ

XCH -DELGMB +1

# -DELGMBY, FETCHING UNLIMITED VALUE

EXTEND

MP SINCDUZ

DDOUBL

COM

EXTEND

# YIELDS +DELATTX UNLIMITD, MAG &lt; 180 DEG.

MSU -DELGMB

# BASED ON UNLIMITED DELGMBV.

TS L

# ONE BIT ERROR IF OPERANDS IN MSU

INDEX NDXCDUW

# OF MIXED SIGNS. WHO CARES?

CA DAXMAX

TC LIMITSUB

TS -DELGMB

# SAVE LIMITED +DELATTX

CCS FLAGOODW

CS -DELGMB

# FETCH IT BACK CHGING SIGN IF WINDOW GOOD

TS -DELGMB

# OTHERWISE USE ZERO FOR -DELATTX

CS -DELGMB +1

EXTEND

MP SINCDUZ

DDOUBL

ADS -DELGMB

# YIELDS -CNTRIB TO -DELATTX FROM -DELGMBY

# -DELGMBX. NO OVERFLOW SINCE LIMITED TO

# 20DEG(1+SIN(70DEG)/COS(70DEG)) &lt; 180DEG

# COMPUTE COMMANDED ATTITUDE RATES

#

# [ OMEGAPD ] [ -2 -4 SINCDUZ +0 ] [ -DELGMBZ ]

# [ ] [ ] [ ]

# [ OMEGAQD ] = [ +0 -8 COSCDUZ COSCDUX -4 SINCDUX ] [ -DELGMBY ]

# [ ] [ ] [ ]

# [ OMEGARD ] [ +0 +8 COSCDUZ SINCDUX -4 COSCDUX ] [ -DELGMBZ ]

#

# ATTITUDE ANGLE RATES IN UNITS OF PI/4 RAD/SEC = K TRIG FCNS IN UNITS OF 2 X GIMBAL ANGLE RATES IN UNITS OF

# PI/2 RAD/SEC. THE CONSTANTS ARE BASED ON DELGMB BEING THE GIMBAL ANGLE CHANGES IN UNITS OF PI RADIANS,

# AND 2 SECONDS BEING THE COMPUTATION PERIOD (THE PERIOD BETWEEN SUCCESSIVE PASSES THRU FINDCDUW).

CS -DELGMB

TS OMEGAPD

CS -DELGMB +1

EXTEND

MP SINCDUZ

DDOUBL

ADS OMEGAPD

ADS OMEGAPD

CS -DELGMB +1

EXTEND

MP COSCDUX

DDOUBL

EXTEND

MP COSCDUZ

TS OMEGAQD

CS -DELGMB +2

EXTEND

MP SINCDUX

ADS OMEGAQD

ADS OMEGAQD

ADS OMEGAQD

CA -DELGMB +1

EXTEND

MP SINCDUX

DDOUBL

EXTEND

MP COSCDUZ

TS OMEGARD

CS -DELGMB +2

EXTEND

MP COSCDUX

ADS OMEGARD

ADS OMEGARD

ADS OMEGARD

# FINAL TRANSFER

|         |        |          |                                          |
|---------|--------|----------|------------------------------------------|
| CDUWXFR | CA     | TWO      |                                          |
|         | TS     | TEM2     |                                          |
|         | INDEX  | TEM2     |                                          |
|         | CA     | -DELGMB  |                                          |
|         | EXTEND |          |                                          |
|         | MP     | DT/DELT  | # RATIO OF DAP INTERVAL TO CDUW INTERVAL |
|         | TC     | ONESTO2S |                                          |
|         | INDEX  | TEM2     |                                          |
|         | TS     | DELCDUX  | # ANGLE INTERFACE                        |
|         | INDEX  | TEM2     |                                          |
|         | CCS    | OMEGAPD  |                                          |
|         | AD     | ONE      |                                          |
|         | TCF    | +2       |                                          |
|         | AD     | ONE      |                                          |
|         | EXTEND |          | # WE NOW HAVE ABS(OMEGAPD,QD,RD)         |
|         | INDEX  | TEM2     |                                          |
|         | MP     | OMEGAPD  |                                          |
|         | EXTEND |          |                                          |
|         | MP     | BIT11    | # 1/16                                   |
|         | EXTEND |          |                                          |
|         | INDEX  | TEM2     | #                                        |
|         | DV     | 1JACC    | # UNITS PI/4 RAD/SEC                     |
|         | TS     | L        |                                          |
|         | CA     | DELERLIM |                                          |
|         | TC     | LIMITSUB |                                          |
|         | INDEX  | TEM2     |                                          |
|         | TS     | DELPEROR | # LAG ANGLE = OMEGA ABS(OMEGA)/2 ACCEL   |
|         | CCS    | TEM2     |                                          |
|         | TCF    | CDUWXFR  |                                          |

# HAUSKEEPING AND RETURN

|         |       |          |                                      |
|---------|-------|----------|--------------------------------------|
| TCQCDUW | CA    | ECDUWUSR |                                      |
|         | TS    | EBANK    | # RETURN USER'S EBANK                |
|         | TC    | INTPRET  |                                      |
|         | SETPD | GOTO     |                                      |
|         |       | 0        |                                      |
|         |       | QCDUWUSR | # NORMAL AND ABNORMAL RETURN TO USER |

# THRUST VECTOR FILTER SUBROUTINE

|         |        |          |                                           |
|---------|--------|----------|-------------------------------------------|
| FLTRSUB | EXTEND |          |                                           |
|         | QXCH   | TEM2     |                                           |
|         | TS     | TEM3     | # SAVE ORIGINAL OFFSET                    |
|         | COM    |          | # ONE MCT, NO WDS, CAN BE SAVED IF NEG OF |
|         | AD     | L        | # ORIG OFFSET ARRIVES IN A, BUT IT'S      |
|         | EXTEND |          | # NOT WORTH THE INCREASED OBSCURITY.      |
|         | INDEX  | NDXCDUW  |                                           |
|         | MP     | GAINFLTR |                                           |
|         | TS     | L        | # INCR TO OFFSET, UNLIMITED               |
|         | CA     | DUNFVLIM | # SAME LIMIT FOR Y AND Z                  |
|         | TC     | LIMITSUB | # YIELDS INCR TO OFFSET, LIMITED          |
|         | AD     | TEM3     | # ORIGINAL OFFSET                         |
|         | TS     | L        | # TOTAL OFFSET, UNLIMITED                 |
|         | CA     | UNFVLIM  | # SAME LIMIT FOR Y AND Z                  |
|         | TC     | LIMITSUB | # YIELDS TOTAL OFFSET, LIMITED            |
|         | TC     | TEM2     |                                           |

# SUBR TO TEST THE ANGLE BETWEEN THE PROPOSED WINDOW AND THRUST CMD VCTS

|          |     |          |                                            |
|----------|-----|----------|--------------------------------------------|
| UNWCTEST | DOT | DSQ      |                                            |
|          |     | UNX/2    |                                            |
|          | DSU | BMN      |                                            |
|          |     | DOTSWFMX |                                            |
|          |     | DCMCL    |                                            |
|          | SSP | RVQ      | # RVQ FOR ALT CHOICE IF DOT MAGN TOO LARGE |
|          |     | FLAGOODW | # ZEROING WINDOW GOOD FLAG                 |
|          |     | 0        |                                            |

# NB2CDUSP RETURNS THE 2'S COMPLEMENT, PI, SP CDU ANGLES X,Y,Z IN MPAC,+1,+2 GIVEN THE MATRIX WHOSE ROW VECTORS  
# ARE THE SEMI-UNIT NAV BASE VECTORS X,Y,X EXPRESSED IN STABLE MEMBER COORDINATES, LOCATED AT 0 IN THE PUSH LIST.  
# NB2CDUSP USES THE ARCTRGSP WHICH HAS A MAXIMUM ERROR OF +-4 BITS.

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| NB2CDUSP | DLOAD  | DSQ      |                                           |
|          |        | 2        |                                           |
|          | BDSU   | BPL      |                                           |
|          |        | DP1/4TH  |                                           |
|          |        | +3       |                                           |
|          | DLOAD  | ZEROVECS | # IN CASE SIN WAS SLIGHTLY > 1/2          |
|          | SQRT   | EXIT     | # YIELDS COS(CDUZ) IN UNITS OF 2          |
|          | EXTEND |          |                                           |
|          | DCA    | MPAC     |                                           |
|          | DDOUBL |          |                                           |
|          | TS     | TEM5     |                                           |
|          | TCF    | +3       |                                           |
|          | CA     | POSMAX   | # OVERFLOW. FETCH POSMAX, MPAC ALWAYS POS |
|          | TS     | TEM5     | # COS(CDUZ) IN TEM5, UNITS 1              |
|          | INDEX  | FIXLOC   |                                           |
|          | CA     | 2        |                                           |
|          | LXCH   | MPAC     |                                           |
|          | TC     | ARCTRGSP |                                           |
|          | TS     | MPAC +2  | # CDUZ                                    |
|          | CA     | ZERO     |                                           |
|          | TC     | DVBYCOSM |                                           |
|          | CA     | FOUR     |                                           |
|          | TC     | DVBYCOSM |                                           |
|          | CS     | TEM1     |                                           |
|          | TC     | ARCTRGSP |                                           |
|          | TS     | MPAC +1  | # CDUY                                    |
|          | CA     | BIT4     |                                           |
|          | TC     | DVBYCOSM |                                           |
|          | CA     | 16OCT    |                                           |
|          | TC     | DVBYCOSM |                                           |
|          | CS     | TEM1     |                                           |
|          | TC     | ARCTRGSP |                                           |
|          | TS     | MPAC     | # CDUX                                    |
|          | TC     | INTPRET  |                                           |
|          | RVQ    |          |                                           |
| 16OCT    | OCT    | 16       |                                           |

```

1 # THE ELEMENTS OF THE NAV BASE MATRIX WHICH WE MUST DIVIDE BY COS(MGA)
2 # ALREADY CONTAIN COS(MGA)/2 AS A FACTOR. THEREFORE THE QUOTIENT SHOULD
3 # ORDINARILY NEVER EXCEED 1/2 IN MAGNITUDE. BUT IF THE MGA IS NEAR PI/2
4 # THEN COS(MGA) IS NEAR ZERO, AND THERE MAY BE SOME CHAFF IN THE OTHER
5 # ELEMENTS OF THE MATRIX WHICH WOULD PRODUCE CHAOS UNDER DIVISION.
6 # BEFORE DIVIDING WE MAKE SURE COS(MGA) IS AT LEAST ONE BIT LARGER
7 # THAN THE MAGNITUDE OF THE HIGH ORDER PART OF THE OPERAND.
8 #
9 # IF ONE OR MORE DIVIDES CANNOT BE PERFORMED, THIS MEANS THAT THE
10 # REQUIRED MGA IS VERY NEARLY +-PI/2 AND THEREFORE THE OTHER GIMBAL
11 # ANGLES ARE INDETERMINATE. THE INNER AND OUTER GIMBAL ANGLES RETURNED
12 # IN THIS CASE WILL BE RANDOM MULTIPLES OF PI/2.
13
14 DVBYCOSM AD FIXLOC
15 TS ADDRWD # ADRES OF OPERAND
16
17 INDEX ADDRWD # FETCH NEG ABS OF OPERAND, AD TEM5, AND
18 CA 0 # SKIP DIVIDE IF RESULT NEG OR ZERO
19 EXTEND
20 BZMF +2
21
22 COM
23 AD TEM5 # C(A) ZERO OR NEG, C(TEM5) ZERO OR POS
24 EXTEND
25 BZMF TSL&TCQ # DIFFERENCE ALWAYS SMALL IF BRANCH
26
27 EXTEND # TEM5 EXCEEDS ABS HIGH ORDER PART OF
28 INDEX ADDRWD # OPERAND BY AT LEAST ONE BIT.
29 DCA 0 # THEREFORE IT EXCEEDS THE DP OPERAND
30 EXTEND # AND DIVISION WILL ALWAYS SUCCEED.
31
32 TSL&TCQ DV TEM5
33 TS L
34 LXCH TEM1
35 TC Q
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

```



# ARCTRGSP RETURNS THE 2'S COMPLEMENT, PI, SP ANGLE IN THE A REGISTER GIVEN ITS SINE IN A AND ITS COSINE IN L IN  
# UNITS OF 2. THE RESULT IS AN UNAMBIGUOUS ANGLE ANYWHERE IN THE CIRCLE, WITH A MAXIMUM ERROR OF +-4 BITS.  
# THE ERROR IS PRODUCED BY THE SUBROUTINE SPARCSIN WHICH IS USED ONLY IN THE REGION +-45 DEGREES.

|          |        |             |                                       |
|----------|--------|-------------|---------------------------------------|
| ARCTRGSP | EXTEND |             |                                       |
|          | BZF    | SINZERO     | # TO AVOID DIVIDING BY ZERO           |
|          | EXTEND |             |                                       |
|          | QXCH   | TEM4        |                                       |
|          | TS     | TEM2        |                                       |
|          | CA     | L           |                                       |
|          | TS     | TEM3        |                                       |
|          | CA     | ZERO        |                                       |
|          | EXTEND |             |                                       |
|          | DV     | TEM2        |                                       |
|          | EXTEND |             |                                       |
|          | BZF    | USECOS      |                                       |
|          | CCS    | TEM3        | # SIN IS SMALLER OR EQUAL             |
|          | CA     | ZERO        |                                       |
|          | TCF    | +4          |                                       |
|          | CS     | TEM2        | # IF COS NEG, REVERSE SIGN OF SIN,    |
|          | TS     | TEM2        | # ANGLE = PI-ARCSIN(SIN)              |
|          | CA     | NEGMAX      | # PICK UP PI, 2'S COMPLEMENT          |
|          | TS     | TEM3        | # WE NO LONGER NEED COS               |
|          | CA     | TEM2        |                                       |
|          | TC     | SPARCSIN -1 |                                       |
|          | TC     | ONESTO2S    |                                       |
|          | EXTEND |             |                                       |
| 1TO2&TCQ | MSU    | TEM3        |                                       |
|          | TC     | ONESTO2S    |                                       |
|          | TC     | TEM4        |                                       |
| USECOS   | CS     | TEM3        | # COS IS SMALLER                      |
|          | TC     | SPARCSIN -1 | # ANGLE = SIGN(SIN)(PI/2-ARCSIN(COS)) |
|          | AD     | HALF        |                                       |
|          | TS     | TEM3        | # WE NO LONGER NEED COS               |
|          | CCS    | TEM2        |                                       |
|          | CA     | TEM3        |                                       |
|          | TCF    | 1TO2&TCQ    |                                       |
|          | CS     | TEM3        |                                       |
|          | TCF    | 1TO2&TCQ    |                                       |
| SINZERO  | CCS    | L           |                                       |
|          | CA     | ZERO        |                                       |
|          | TC     | Q           |                                       |
|          | CA     | NEGMAX      | # PI, 2'S COMP                        |
|          | TC     | Q           |                                       |

# FINDCDUW--GUIDAP\_INTERFACE

# SPARCSIN TAKES AN ARGUMENT SCALED UNITY IN A AND RETURNS AN ANGLE SCALED  
# 180 DEGREES IN A. IT HAS BEEN UNIT TESTED IN THE REGION +-.94 (+-70  
# DEGREES) AND THE MAXIMUM ERROR IS +-5 BITS WITH AN AVERAGE TIME OF  
# 450 MICROSECONDS. SPARCSIN -1 TAKES THE ARGUMENT SCALED TWO. (BOB CRISP)

|          |        |        |
|----------|--------|--------|
| SPARCSIN | DOUBLE |        |
|          | TS     | SR     |
|          | TCF    | +4     |
|          | INDEX  | A      |
|          | CS     | LIMITS |
|          | TS     | SR     |
|          | EXTEND |        |
|          | MP     | A      |
|          | TS     | TEM1   |
|          | EXTEND |        |
|          | MP     | DPL9   |
|          | AD     | DPL7   |
|          | EXTEND |        |
|          | MP     | TEM1   |
|          | AD     | DPL5   |
|          | EXTEND |        |
|          | MP     | TEM1   |
|          | AD     | DPL3   |
|          | EXTEND |        |
|          | MP     | TEM1   |
|          | AD     | DPL1   |
|          | EXTEND |        |
|          | MP     | SR     |
|          | TC     | Q      |
| DPL1     | DEC    | 10502  |
| DPL3     | DEC    | 432    |
| DPL5     | DEC    | 7300   |
| DPL7     | DEC    | -11803 |
| DPL9     | DEC    | 8397   |

# LIMITSUB LIMITS THE MAGNITUDE OF THE POSITIVE OR NEGATIVE VARIABLE  
# ARRIVING IN L TO THE POSITIVE LIMIT ARRIVING IN A.  
# THE SIGNED LIMITED VARIABLE IS RETURNED IN A.  
#  
# VERSION COUTESY HUGH BLAIR-SMITH

|          |        |      |
|----------|--------|------|
| LIMITSUB | TS     | TEM1 |
|          | CA     | ZERO |
|          | EXTEND |      |
|          | DV     | TEM1 |
|          | CCS    | A    |
|          | LXCH   | TEM1 |
|          | TCF    | +2   |
|          | TCF    | +3   |
|          | CA     | L    |
|          | TC     | Q    |
|          | CS     | TEM1 |
|          | TC     | Q    |

# SUBROUTINE TO CONVERT 1'S COMP SP TO 2'S COMP

|          |     |     |
|----------|-----|-----|
| ONESTO2S | CCS | A   |
|          | AD  | ONE |
|          | TC  | Q   |
|          | CS  | A   |
|          | TC  | Q   |

# NO ATTITUDE CONTROL

|          |     |       |
|----------|-----|-------|
| NOATTCNT | TC  | ALARM |
|          | OCT | 00402 |

# NO ATTITUDE CONTROL

|    |        |          |                                          |
|----|--------|----------|------------------------------------------|
| +2 | INHINT |          | # COME HERE FOR NOATTCNT WITHOUT ALARM   |
|    | TC     | IBNKCALL | # RELINT AT TC INTPRET AFTER TCQCDUW     |
|    | FCADR  | STOPRATE |                                          |
|    | TCF    | TCQCDUW  | # RETURN TO USER SKIPPING AUTOPILOT CMDS |

# MIDDLE GIMBAL ANGLE ALARM

|          |     |        |
|----------|-----|--------|
| ALARMMGA | TC  | ALARM  |
|          | OCT | 00401  |
|          | TCF | MGARET |

\*\*\*\*\*

## # CONSTANTS

\*\*\*\*\*

## # ADDRESS CONSTANTS

ECDUWL            ECADR    ECDUW

## # THRUST DIRECTION FILTER CONSTANTS

GAINFLTR        DEC        .2            # GAIN FILTER SANS CSM

DEC            .1            # GAIN FILTER WITH CSM

DUNFVLIM        DEC        .007 B-1        # 7 MR MAX CHG IN F DIR IN VEH IN 2 SECS.

# THIS DOES NOT ALLOW FOR S/C ROT RATE.

UNFVLIM        DEC        .129 B-1        # 129 MR MAX THRUST OFFSET. 105 MR TRAVEL

# +10MR DEFL+5MR MECH MOUNT+9MR ABLATION.

## # CONSTANT RELATED TO GIMBAL ANGLE COMPUTATIONS

DOTSWFMX        DEC        .93302 B-4        # LIM COLNRTY OF UNWC/2 &amp; UNFC/2 TO 85 DEG

# LOWER PART COMES FROM NEXT CONSTANT

DAXMAX        DEC        .1111111111        # DELATTX LIM TO 20 DEG IN 2 SECS, 1'S, PI

DEC            .0111111111        # 2 DEG WHEN CSM DOCKED

DAY/2MAX        DEC        .0555555555        # LIKEWISE FOR DELATTY

DEC            .0055555555

DAZMAX        =        DAXMAX        # LIKEWISE FOR DELATTZ

CDUZDLIM        DEC        .3888888888        # 70 DEG LIMIT FOR MGA, 1'S, PI

## # CONSTANTS FOR DATA TRANSFER

DT/DELT        DEC        .05            # .1 SEC/2 SEC WHICH IS THE AUTOPILOT

# CONTROL SAMPLE PERIOD/COMPUTATION PERIOD

DELERLIM        =        DAY/2MAX        #        0 DEG LIMIT FOR LAG ANGLES, 1'S, PI

# \*\*\* END OF FLY        .132 \*\*\*

```
1 # PROGRAM NAME -- PROG52 DATE -- JAN 9, 1967
2 # MOD NO -- 0 LOG SECTION -- P51-P53
3 # MODIFICATION BY -- LONSKE ASSEMBLY -- SUNDANCE REV 46
4 #
5 # FUNCTIONAL DESCRIPTION --
6 #
7 # ALIGNS THE IMU TO ONE OF THREE ORIENTATIONS SELECTED BY THE ASTRONAUT. THE PRESENT IMU ORIENTATION IS KNOWN
8 # AND IS STORED IN REFSMMAT. THE THREE POSSIBLE ORIENTATIONS MAY BE:
9 #
10 # (A) PREFERRED ORIENTATION
11 #
12 # AN OPTIMUM ORIENTATION FOR A PREVIOUSLY CALCULATED MANEUVER. THIS ORIENTATION MUST BE CALCULATED AND
13 # STORED BY A PREVIOUSLY SELECTED PROGRAM.
14 #
15 # (B) NOMINAL ORIENTATION
16 #
17 # X = UNIT (R)
18 # -SM
19 # Y = UNIT (V X R)
20 # SM
21 # Z = UNIT (X X Y)
22 # SM SM SM
23 #
24 # WHERE:
25 #
26 # R = THE GEOCENTRIC RADIUS VECTOR AT TIME T(ALIGN) SELECTED BY THE ASTRONAUT
27 # -
28 #
29 # V = THE INERTIAL VELOCITY VECTOR AT TIME T(ALIGN) SELECTED BY THE ASTRONAUT
30 # -
31 #
32 # (C) RERSMMAT ORIENTATION
33 #
34 # (D) LANDING SITE -- THIS IS NOT AVAILABLE IN SUNDANCE
35 #
36 # THIS SELECTION CORRECTS THE PRESENT IMU ORIENTATION. THE PRESENT ORIENTATION DIFFERS FROM THAT TO WHICH IT
37 # WAS LAST ALIGNED ONLY DUE TO GYRO DRIFT (I.E., NEITHER GIMBAL LOCK NOR IMU POWER INTERRUPTION HAS OCCURRED
38 # SINCE THE LAST ALIGNMENT).
39 #
40 # AFTER A IMU ORIENTATION HAS BEEN SELECTED ROUTINE S52.2 IS OPERATED TO COMPUTE THE GIMBAL ANGLES USING THE
41 # NEW ORIENTATION AND THE PRESENT VEHICLE ATTITUDE. CAL52A THEN USES THESE ANGLES, STORED IN THETAD,+1,+2, TO
42 # COARSE ALIGN THE IMU. THE STARS SELECTION ROUTINE, R56, IS THEN OPERATED. IF 2 STARS ARE NOT AVAILABLE AN ALARM
43 # IS FLASHED TO NOTIFY THE ASTRONAUT. AT THIS POINT THE ASTRONAUT WILL MANEUVER THE VEHICLE AND SELECT 2 STARS
44 # EITHER MANUALLY OR AUTOMATICALLY. AFTER 2 STARS HAVE BEEN SELECTED THE IMU IS FINE ALIGNED USING ROUTINE R51. IF
45 # THE RENDEZVOUS NAVIGATION PROCESS IS OPERATING (INDICATED BY RNDVZFLG) P20 IS DISPLAYED. OTHERWISE P00 IS
46 # REQUESTED.
47 #
48 # CALLING SEQUENCE --
49 #
```

# THE PROGRAM IS CALLED BY THE ASTRONAUT BY DSKY ENTRY.

# SUBROUTINES CALLED --

|   |             |             |              |
|---|-------------|-------------|--------------|
| # | 1. FLAGDOWN | 7. S52.2    | 13. NEWMODEX |
| # | 2. R02BOTH  | 8. CAL53A   | 14. PRIOLARM |
| # | 3. GOPERF4  | 9. FLAGUP   |              |
| # | 4. MATMOVE  | 10. R56     |              |
| # | 5. GOFLASH  | 11. R51     |              |
| # | 6. S52.3    | 12. GOPERF3 |              |

# NORMAL EXIT MODES --

# EXITS TO ENDOFJOB

# ALARM OR ABORT EXIT MODES --

# NONE

# OUTPUT --

# THE FOLLOWING MAY BE FLASHED ON THE DSKY

|   |                                                              |
|---|--------------------------------------------------------------|
| # | 1. IMU ORIENTATION CODE                                      |
| # | 2. ALARM CODE 215 -- PREFERRED IMU ORIENTATION NOT SPECIFIED |
| # | 3. TIME OF NEXT IGNITION                                     |
| # | 4. GIMBAL ANGLES                                             |
| # | 5. ALARM CODE 405 -- TWO STARS NOT AVAILABLE                 |
| # | 6. PLEASE PERFORM P00                                        |

# THE MODE DISPLAY MAY BE CHANGED TO 20

# ERASABLE INITIALIZATION REQUIRED --

# PFRATFLG SHOULD BE SET IF A PREFERRED ORIENTATION HAS BEEN COMPUTED. IF IT HAS BEEN COMPUTED IT IS STORED IN XSMD, YSMD, ZSMD.

# RNDVZFLG INDICATES WHETHER THE RENDEZVOUS NAVIGATION PROCESS IS OPERATING.

# DEBRIS --

# WORK AREA

|  |        |      |
|--|--------|------|
|  | BANK   | 33   |
|  | SETLOC | P50S |
|  | BANK   |      |

|  |        |          |
|--|--------|----------|
|  | EBANK= | BESTI    |
|  | COUNT* | \$\$/P52 |

|        |      |          |                    |
|--------|------|----------|--------------------|
| PROG52 | TC   | BANKCALL |                    |
|        | CADR | R02BOTH  | # IMU STATUS CHECK |
|        | CAF  | PFRATBIT |                    |

|  |      |          |                    |
|--|------|----------|--------------------|
|  | MASK | FLAGWRD2 | # IS PFRATFLG SET? |
|  | CCS  | A        |                    |

|      |        |          |                                          |
|------|--------|----------|------------------------------------------|
|      | TC     | P52A     | # YES                                    |
|      | CAF    | BIT2     | # NO                                     |
| P52A | TC     | P52A +1  |                                          |
|      | CAF    | BIT1     |                                          |
|      | TS     | OPTION2  |                                          |
| P52B | CAF    | BIT1     |                                          |
|      | TC     | BANKCALL | # FLASH OPTION CODE AND ORIENTATION CODE |
|      | CADR   | GOPERF4R | # FLASH V04N06                           |
|      | TC     | GOTOP00H |                                          |
|      | TCF    | +5       | # V33 -- PROCEED                         |
|      | TC     | P52B     | # NEW CODE -- NEW ORIENTATION CODE INPUT |
|      | TC     | PHASCHNG | # DISPLAY RETURN                         |
|      | OCT    | 00014    |                                          |
|      | TC     | ENDOFJOB |                                          |
|      | CA     | OPTION2  |                                          |
|      | MASK   | THREE    |                                          |
|      | INDEX  | A        |                                          |
|      | TC     | +1       |                                          |
|      | TCF    | OPT4     | # OPTION 4 LANDING SITE                  |
|      | TCF    | P52H     | # OPTION 1 PREFERRED                     |
|      | TCF    | P52T     | # OPTION 2 NOMINAL                       |
| P52E | TC     | INTPRET  | # OPTION 3 REFSMMAT                      |
|      | GOTO   |          |                                          |
|      |        | P52F     | # GO DO R51                              |
| OPT4 | EXTEND |          |                                          |
|      | DCA    | TLAND    | # IF OPTION 4 DISPLAY TLAND              |
|      | TCF    | P52T +2  |                                          |
| P52T | EXTEND |          |                                          |
|      | DCA    | NEG0     |                                          |
|      | DXCH   | DSPTM1   |                                          |
|      | CAF    | V06N34*  |                                          |
|      | TC     | BANKCALL |                                          |
|      | CADR   | GOFLASH  |                                          |
|      | TC     | GOTOP00H |                                          |
|      | TC     | +2       |                                          |
|      | TC     | -5       |                                          |
|      | DXCH   | DSPTM1   |                                          |
|      | EXTEND |          |                                          |
|      | BZMF   | +2       | # IF TIME ZERO OR NEG USE TIME2          |
|      | TCF    | +3       |                                          |
|      | EXTEND |          |                                          |
|      | DCA    | TIME2    |                                          |
|      | DXCH   | TALIGN   |                                          |
| P52V | CA     | OPTION2  |                                          |
|      | MASK   | BIT2     |                                          |
|      | CCS    | A        |                                          |
|      | TC     | P52W     |                                          |



|    |  |  |  |    |
|----|--|--|--|----|
| 1  |  |  |  | 1  |
| 2  |  |  |  | 2  |
| 3  |  |  |  | 3  |
| 4  |  |  |  | 4  |
| 5  |  |  |  | 5  |
| 6  |  |  |  | 6  |
| 7  |  |  |  | 7  |
| 8  |  |  |  | 8  |
| 9  |  |  |  | 9  |
| 10 |  |  |  | 10 |
| 11 |  |  |  | 11 |
| 12 |  |  |  | 12 |
| 13 |  |  |  | 13 |
| 14 |  |  |  | 14 |
| 15 |  |  |  | 15 |
| 16 |  |  |  | 16 |
| 17 |  |  |  | 17 |
| 18 |  |  |  | 18 |
| 19 |  |  |  | 19 |
| 20 |  |  |  | 20 |
| 21 |  |  |  | 21 |
| 22 |  |  |  | 22 |
| 23 |  |  |  | 23 |
| 24 |  |  |  | 24 |
| 25 |  |  |  | 25 |
| 26 |  |  |  | 26 |
| 27 |  |  |  | 27 |
| 28 |  |  |  | 28 |
| 29 |  |  |  | 29 |
| 30 |  |  |  | 30 |
| 31 |  |  |  | 31 |
| 32 |  |  |  | 32 |
| 33 |  |  |  | 33 |
| 34 |  |  |  | 34 |
| 35 |  |  |  | 35 |
| 36 |  |  |  | 36 |
| 37 |  |  |  | 37 |
| 38 |  |  |  | 38 |
| 39 |  |  |  | 39 |
| 40 |  |  |  | 40 |
| 41 |  |  |  | 41 |
| 42 |  |  |  | 42 |
| 43 |  |  |  | 43 |
| 44 |  |  |  | 44 |
| 45 |  |  |  | 45 |
| 46 |  |  |  | 46 |
| 47 |  |  |  | 47 |
| 48 |  |  |  | 48 |
| 49 |  |  |  | 49 |
| 50 |  |  |  | 50 |
| 51 |  |  |  | 51 |
| 52 |  |  |  | 52 |
| 53 |  |  |  | 53 |
| 54 |  |  |  | 54 |
| 55 |  |  |  | 55 |
| 56 |  |  |  | 56 |
| 57 |  |  |  | 57 |
| 58 |  |  |  | 58 |
| 59 |  |  |  | 59 |
| 60 |  |  |  | 60 |

TC           INTPRET           # OPTION 4 -- GET LS ORIENTATION  
GOTO           P52LS





|    |          |       |          |                                             |    |
|----|----------|-------|----------|---------------------------------------------|----|
| 1  |          |       |          |                                             | 1  |
| 2  | P52W     | TC    | INTPRET  |                                             | 2  |
| 3  |          | DLOAD | CALL     | # PICK UP ALIGN TIME                        | 3  |
| 4  |          |       | TALIGN   | # COMPUTED NOMINAL IMU                      | 4  |
| 5  |          |       | S52.3    | # ORIENTATION                               | 5  |
| 6  | P52D     | CALL  |          | # READ VEHICLE ATTITUDE AND                 | 6  |
| 7  |          |       | S52.2    | # COMPUTE GIMBAL ANGLES                     | 7  |
| 8  |          | EXIT  |          |                                             | 8  |
| 9  |          | CAF   | V06N22   |                                             | 9  |
| 10 |          | TC    | BANKCALL | # DISPLAY GIMBAL ANGLES                     | 10 |
| 11 |          | CADR  | GOFLASH  |                                             | 11 |
| 12 |          | TC    | GOTOP00H |                                             | 12 |
| 13 | P52H     | TCF   | COARSTYP | # V33 -- PROCEED, SEE IF GYRO TORQUE COARSE | 13 |
| 14 |          | TC    | INTPRET  |                                             | 14 |
| 15 |          | GOTO  |          |                                             | 15 |
| 16 |          |       | P52D     |                                             | 16 |
| 17 | REGCOARS | TC    | INTPRET  |                                             | 17 |
| 18 |          | CALL  |          | # DO COARSE ALIGN                           | 18 |
| 19 |          |       |          | # ROUTINE                                   | 19 |
| 20 | COARSRET | SET   | CAL53A   |                                             | 20 |
| 21 |          |       | CLEAR    |                                             | 21 |
| 22 |          |       | REFSMFLG |                                             | 22 |
| 23 | P52F     | CALL  | PFRATFLG |                                             | 23 |
| 24 |          |       | R51      |                                             | 24 |
| 25 | P52OUT   | EXIT  |          |                                             | 25 |
| 26 |          | TC    | GOTOP00H |                                             | 26 |
| 27 | VB05N09  | =     | V05N09   |                                             | 27 |
| 28 | V06N34*  | VN    | 634      |                                             | 28 |
| 29 |          |       |          |                                             | 29 |
| 30 |          |       |          |                                             | 30 |
| 31 |          |       |          |                                             | 31 |
| 32 |          |       |          |                                             | 32 |
| 33 |          |       |          |                                             | 33 |
| 34 |          |       |          |                                             | 34 |
| 35 |          |       |          |                                             | 35 |
| 36 |          |       |          |                                             | 36 |
| 37 |          |       |          |                                             | 37 |
| 38 |          |       |          |                                             | 38 |
| 39 |          |       |          |                                             | 39 |
| 40 |          |       |          |                                             | 40 |
| 41 |          |       |          |                                             | 41 |
| 42 |          |       |          |                                             | 42 |
| 43 |          |       |          |                                             | 43 |
| 44 |          |       |          |                                             | 44 |
| 45 |          |       |          |                                             | 45 |
| 46 |          |       |          |                                             | 46 |
| 47 |          |       |          |                                             | 47 |
| 48 |          |       |          |                                             | 48 |
| 49 |          |       |          |                                             | 49 |
| 50 |          |       |          |                                             | 50 |
| 51 |          |       |          |                                             | 51 |
| 52 |          |       |          |                                             | 52 |
| 53 |          |       |          |                                             | 53 |
| 54 |          |       |          |                                             | 54 |
| 55 |          |       |          |                                             | 55 |
| 56 |          |       |          |                                             | 56 |
| 57 |          |       |          |                                             | 57 |
| 58 |          |       |          |                                             | 58 |
| 59 |          |       |          |                                             | 59 |
| 60 |          |       |          |                                             | 60 |

# P51-P53

# CHECK FOR GYRO TORQUE COARSE ALIGNMENT

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| COARSTYP | CAF    | OCT13    |                                            |
|          | TC     | BANKCALL | # DISPLAY V 50N25 WITH COARSE ALIGN OPTION |
|          | CADR   | GOPERF1  |                                            |
|          | TCF    | GOTOPOOH | # V34 -- TERMIN&OE                         |
|          | TCF    | REGCOARS | # V33 -- NORMAL COARSE                     |
|          | TC     | INTPRET  | # V32 -- GYRO TORQUE COARSE                |
|          | VLOAD  | MXV      |                                            |
|          |        | XSMD     | # GET SM(DESIRED) WRT SM(PRESENT)          |
|          |        | REFSMMAT |                                            |
|          | UNIT   |          |                                            |
|          | STOVL  | XDC      |                                            |
|          |        | YSMD     |                                            |
|          | MXV    | UNIT     |                                            |
|          |        | REFSMMAT |                                            |
|          | STOVL  | YDC      |                                            |
|          |        | ZSMD     |                                            |
|          | MXV    | UNIT     |                                            |
|          |        | REFSMMAT |                                            |
|          | STCALL | ZDC      |                                            |
|          |        | GYCOARS  |                                            |
|          | GOTO   |          |                                            |
| OCT13    | OCT    | P52OUT   |                                            |
|          |        | 13       |                                            |

1412THE  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
1

# P51-P53

# COMPUTE LANDING ORIENTATION FOR OPTION 4

|       |        |          |                                        |
|-------|--------|----------|----------------------------------------|
| P52LS | SET    | CLEAR    | # GET LANDING SITE ORIENTATION         |
|       |        | LUNAFLAG |                                        |
|       |        | ERADFLAG | # TO PICK UP RLS                       |
|       | SETPD  | VLOAD    |                                        |
|       |        | 0        |                                        |
|       |        | RLS      | # PICK UP LANDING SITE VEC IN MF       |
|       | PDDL   | PUSH     | # RLS PD 0-5                           |
|       |        | TALIGN   |                                        |
|       | STCALL | TLAND    | # JAM ALIGN TIME IN TLAND FOR OPTION 4 |
|       |        | RP-TO-R  | # TRANS RLS TO REF                     |
|       | VSR2   |          |                                        |
|       | STODL  | ALPHAV   | # INPUT TO LAT-LONG                    |
|       |        | TALIGN   |                                        |
|       | CALL   |          |                                        |
|       |        | N89DISP  |                                        |
|       | VLOAD  | UNIT     | # COMPUTE LANDING SITE ORIENT (XSMD)   |
|       |        | ALPHAV   |                                        |
|       | STCALL | XSMD     |                                        |
|       |        | LSORIENT |                                        |
|       | GOTO   |          |                                        |
|       |        | P52D     | # NOW GO COMPUTE GIMBAL ANGLES.        |

1412THE

# SUBROUTINE TO CALCULATE AND DISPLAY THE LUNAR LANDING SITE

SETLOC P50S1  
BANK  
EBANK= XSM

N89DISP

STQ

QMAJ

STCALL GDT/2 +4 # TEMP STORE TIME

LAT-LONG

DLOAD SR1

LONG

STODL LANDLONG

ALT

STODL LANDALT

LAT

STODL LANDLAT

EXIT

LSDISP

CAF V06N89\* # DISPLAY LAT, LONG/2, ALT

TC BANKCALL

CADR GOFLASH

TCF GOTOP00H # V34 -- TERMINATE -- EXIT P57

TCF +2 # V33 -- PROCEED -- ACCEPT LS DATA

TCF LSDISP # V32 OR E -- LOOK AGAIN AND/OR LOAD NEW LS

TC INTPRET

DLOAD SL1

LANDLONG

STODL LONG

LANDALT

STODL ALT

LANDLAT

STODL LAT

GDT/2 +4 # PICK UP TIME

CALL # GET RLS BACK FROM LAT, LONG, ALT

GOTO # RLS B-29 IN MPAC AND ALPHAV

V06N89\*

VN

QMAJ  
689

# P51-P53

```
NAME -- S50 ALIAS LOCSAM
BY
VINCENT
FUNCTION -- COMPUTE INPUTS FOR PICAPAR AND PLANET

#
DEFINE
#
U = UNIT(SUN WRT EARTH)
ES
#
U = UNIT(MOON WRT EARTH)
EM
#
R = POSITION VECTOR OF LEM
L
#
R = MEAN DISTANCE (384402KM) BETWEEN EARTH AND MOON
EM
#
P = RATIO R /(DISTANCE SUN TO EARTH) > .00257125
EM
#
R = EQUATORIAL RADIUS (6378.166KM) OF EARTH
E
#
LOCSAM COMPUTES IN EARTH INFLUENCE
#
VSUN = U
ES
#
VEARTH = -UNIT(R)
L
#
VMOON = UNIT(R .U - R)
EM EM L
#
CSUN = COS 90
#
CEARTH = COS(5 + ARCSIN(R /MAG(R)))
E L
#
CMOON = COS 5
#
INPUT -- TIME IN MPAC
#
OUTPUT -- LISTED ABOVE
#
SUBROUTINES -- LSPOS, LEMPREC
#
DEBRIS -- VAC AREA, TSIGHT
```

# P51-P53

|    |          |        |            |    |
|----|----------|--------|------------|----|
| 1  |          |        |            | 1  |
| 2  |          | COUNT* | \$\$/LOSAM | 2  |
| 3  |          |        |            | 3  |
| 4  | S50      | =      | LOCSAM     | 4  |
| 5  | LOCSAM   | STQ    |            | 5  |
| 6  |          |        | QMIN       | 6  |
| 7  |          | STCALL | TSIGHT     | 7  |
| 8  |          |        | LSPOS      | 8  |
| 9  |          | DLOAD  |            | 9  |
| 10 |          |        | TSIGHT     | 10 |
| 11 |          | STCALL | TDEC1      | 11 |
| 12 |          |        | LEMPREC    | 12 |
| 13 |          | SSP    | TIX,2      | 13 |
| 14 |          |        | S2         | 14 |
| 15 |          |        | 0          | 15 |
| 16 |          |        | MOONCNTR   | 16 |
| 17 | EARTCNTR | VLOAD  | VXSC       | 17 |
| 18 |          |        | VMOON      | 18 |
| 19 |          |        | RSUBEM     | 19 |
| 20 |          | VSL1   | VSU        | 20 |
| 21 |          |        | RATT       | 21 |
| 22 |          | UNIT   |            | 22 |
| 23 |          | STOVL  | VMOON      | 23 |
| 24 |          |        | RATT       | 24 |
| 25 |          | UNIT   | VCOMP      | 25 |
| 26 |          | STODL  | VEARTH     | 26 |
| 27 |          |        | RSUBE      | 27 |
| 28 |          | CALL   |            | 28 |
| 29 |          |        | OCCOS      | 29 |
| 30 |          | STODL  | CEARTH     | 30 |
| 31 |          |        | CSS5       | 31 |
| 32 |          | STCALL | CMOON      | 32 |
| 33 |          |        | ENDSAM     | 33 |
| 34 | MOONCNTR | VLOAD  | VXSC       | 34 |
| 35 |          |        | VMOON      | 35 |
| 36 |          |        | ROE        | 36 |
| 37 |          | BVSU   | UNIT       | 37 |
| 38 |          |        | VSUN       | 38 |
| 39 |          | STOVL  | VSUN       | 39 |
| 40 |          |        | VMOON      | 40 |
| 41 |          | VXSC   | VAD        | 41 |
| 42 |          |        | RSUBEM     | 42 |
| 43 |          |        | RATT       | 43 |
| 44 |          | UNIT   | VCOMP      | 44 |
| 45 |          | STOVL  | VEARTH     | 45 |
| 46 |          |        | RATT       | 46 |
| 47 |          | UNIT   | VCOMP      | 47 |
| 48 |          | STODL  | VMOON      | 48 |
| 49 |          |        | RSUBM      | 49 |
| 50 |          | CALL   |            | 50 |
| 51 |          |        | OCCOS      | 51 |
| 52 |          |        |            | 52 |
| 53 |          |        |            | 53 |
| 54 |          |        |            | 54 |
| 55 |          |        |            | 55 |
| 56 |          |        |            | 56 |
| 57 |          |        |            | 57 |
| 58 |          |        |            | 58 |
| 59 |          |        |            | 59 |
| 60 |          |        |            | 60 |



```
1 # PROGRAM NAME -- R56 DATE: DEC 20 66
2 # MOD 1 LOG SECTION: P51-P53
3 # ASSEMBLY: SUNDISK REV4D
4 # BY KEN VINCENT
5 #
6 # FUNCTION
7 # THIS PROGRAM READS THE IMU-CDUS AND COMPUTES THE VEHICLE ORIENTATION
8 # WITH RESPECT TO INERTIAL SPACE. IT THEN COMPUTES THE SHAFT AXIS (SAX)
9 # WITH RESPECT TO REFERENCE INTERTIAL. EACH STAR IN THE CATALOG IS TESTED
10 # TO DETERMIN IF IT IS OCCULTED BY EITHER EARTH, SUN OR MOON. IF A
11 # STAR IS NOT OCCULTED THEN IT IS PAIRED WITH ALL STARS OF LOWER INDEX.
12 # THE PAIRED STAR IS TESTED FOR OCCULTATION. PAIRS OF STARS THAT PASS
13 # THE OCCULTATION TESTS ARE TESTED FOR GOOD SEPARATION. A PAIR OF STARS
14 # HAVE GOOD SEPARATION IF THE ANGLE BETWEEN THEM IS LESS THAN 100 DEGREES
15 # AND MORE THAN 50 DEGREES. THOSE PAIRS WITH GOOD SEPARATION
16 # ARE THEN TESTED TO SEE IF THEY LIE IN CURRENT FIELD OF VIEW. (WITHIN
17 # 50 DEGREES OF SAX). THE PAIR WITH MAX SEPARATION IS CHOSEN FROM
18 # THOSE WITH GOOD SEPARATION, AND IN FIELD OF VIEW.
19 #
20 # CALLING SEQUENCE
21 # L TC BANKCALL
22 # L+1 CADR R56
23 # L+2 ERROR RETURN -- NO STARS IN FIELD OF VIEW
24 # L+3 NORMAL RETURN
25 #
26 # OUTPUT
27 # BESTI, BESTJ -- SINGLE PREC, INTEGERS, STAR NUMBERS TIMES 6
28 # VFLAG -- FLAG BIT SET IMPLIES NO STARS IN FIELD OF VIEW
29 #
30 # INITIALIZATION
31 # 1) A CALL TO LOCSAM MUST BE MADE
32 #
33 # DEBRIS
34 # WORKAREA
35 # X,Y,ZNB
36 # SINCDU, COSCDU
37 # STARAD -- STAR +5
38 #
39
40 R56 = PICAPAR
41 COUNT* $$/R56
42 PICAPAR TC MAKECADR
43 TS QMIN
44 TC INTPRET
45 CALL
46 CDUTRIG
47 CALL
48 CALCSMSC
49 SETPD
50 0
51 SET DLOAD # VFLAG = 1
52 VFLAG
```



|    |           |        |                    |    |
|----|-----------|--------|--------------------|----|
| 1  | # 151 155 |        | PAGE 150           | 1  |
| 2  |           |        | DPZERO             | 2  |
| 3  |           | STOVL  | BESTI              | 3  |
| 4  |           |        | XNB                | 5  |
| 5  |           | VXSC   | PDVL               | 6  |
| 6  |           |        | HALFDP             | 7  |
| 7  |           |        | ZNB                | 9  |
| 8  |           | AXT,1  | VXSC               | 10 |
| 9  |           |        | 228D               | 11 |
| 10 |           |        | # X1 = 37 X 6 + 6  | 12 |
| 11 |           |        | HALFDP             | 13 |
| 12 |           | VAD    |                    | 14 |
| 13 |           | VXM    | UNIT               | 15 |
| 14 |           |        | REFSMMAT           | 16 |
| 15 |           | STORE  | SAX                | 17 |
| 16 |           | SSP    | # SAX = SHAFT AXIS | 18 |
| 17 |           |        | # S1 = S2 = 6      | 19 |
| 18 |           |        | S1                 | 20 |
| 19 |           |        | 6                  | 21 |
| 20 |           |        | S2                 | 22 |
| 21 |           |        | 6                  | 23 |
| 22 | PIC1      | TIX,1  | GOTO               | 24 |
| 23 |           |        | # MAJOR STAR       | 25 |
| 24 |           |        | PIC2               | 26 |
| 25 |           |        | PICEND             | 27 |
| 26 | PIC2      | VLOAD* | DOT                | 28 |
| 27 |           |        | CATLOG,1           | 29 |
| 28 |           |        | SAX                | 30 |
| 29 |           | DSU    | BMN                | 31 |
| 30 |           |        | CSS33              | 32 |
| 31 |           |        | PIC1               | 33 |
| 32 |           | LXA,2  |                    | 34 |
| 33 |           |        | X1                 | 35 |
| 34 | PIC3      | TIX,2  | GOTO               | 36 |
| 35 |           |        | PIC4               | 37 |
| 36 |           |        | PIC1               | 38 |
| 37 | PIC4      | VLOAD* | DOT                | 39 |
| 38 |           |        | CATLOG,2           | 40 |
| 39 |           |        | SAX                | 41 |
| 40 |           | DSU    | BMN                | 42 |
| 41 |           |        | CSS33              | 43 |
| 42 |           |        | PIC3               | 44 |
| 43 |           | VLOAD* | DOT*               | 45 |
| 44 |           |        | CATLOG,1           | 46 |
| 45 |           |        | CATLOG,2           | 47 |
| 46 |           | DSU    | BPL                | 48 |
| 47 |           |        | CSS40              | 49 |
| 48 |           |        | PIC3               | 50 |
| 49 |           | VLOAD* | CALL               | 51 |
| 50 |           |        | CATLOG,1           | 52 |
| 51 |           |        | OCCULT             | 53 |
| 52 |           | BON    |                    | 54 |
| 53 |           |        | CULTFLAG           | 55 |
| 54 |           |        | PIC1               | 56 |
| 55 |           |        |                    | 57 |
| 56 |           |        |                    | 58 |
| 57 |           |        |                    | 59 |
| 58 |           |        |                    | 60 |
| 59 |           |        |                    | 61 |
| 60 |           |        |                    | 62 |



|    |        |       |          |    |
|----|--------|-------|----------|----|
| 1  |        |       |          | 1  |
| 2  |        |       |          | 2  |
| 3  |        |       |          | 3  |
| 4  | PICGXT | TC    | VFLAG    | 4  |
| 5  |        | LXA,1 | PICGXT   | 5  |
| 6  |        |       | PICBXT   | 6  |
| 7  |        |       | LXA,2    | 7  |
| 8  |        |       | BESTI    | 8  |
| 9  |        |       | BESTJ    | 9  |
| 10 | VLOAD  |       | DOT*     | 10 |
| 11 |        |       | SAX      | 11 |
| 12 |        |       | CATLOG,1 | 12 |
| 13 |        |       | DOT*     | 13 |
| 14 |        |       | SAX      | 14 |
| 15 |        |       | CATLOG,2 | 15 |
| 16 |        |       |          | 16 |
| 17 |        |       |          | 17 |
| 18 |        |       |          | 18 |
| 19 |        |       |          | 19 |
| 20 |        |       |          | 20 |
| 21 |        |       |          | 21 |
| 22 |        |       |          | 22 |
| 23 |        |       |          | 23 |
| 24 |        |       |          | 24 |
| 25 |        |       |          | 25 |
| 26 |        |       |          | 26 |
| 27 |        |       |          | 27 |
| 28 |        |       |          | 28 |
| 29 |        |       |          | 29 |
| 30 |        |       |          | 30 |
| 31 |        |       |          | 31 |
| 32 |        |       |          | 32 |
| 33 |        |       |          | 33 |
| 34 |        |       |          | 34 |
| 35 |        |       |          | 35 |
| 36 |        |       |          | 36 |
| 37 |        |       |          | 37 |
| 38 |        |       |          | 38 |
| 39 |        |       |          | 39 |
| 40 |        |       |          | 40 |
| 41 |        |       |          | 41 |
| 42 |        |       |          | 42 |
| 43 |        |       |          | 43 |
| 44 |        |       |          | 44 |
| 45 |        |       |          | 45 |
| 46 |        |       |          | 46 |
| 47 |        |       |          | 47 |
| 48 |        |       |          | 48 |
| 49 |        |       |          | 49 |
| 50 |        |       |          | 50 |
| 51 |        |       |          | 51 |
| 52 |        |       |          | 52 |
| 53 |        |       |          | 53 |
| 54 |        |       |          | 54 |
| 55 |        |       |          | 55 |
| 56 |        |       |          | 56 |
| 57 |        |       |          | 57 |
| 58 |        |       |          | 58 |
| 59 |        |       |          | 59 |
| 60 |        |       |          | 60 |

```

1 # NAME -- R51 FINE ALIGN
2 # FUNCTION -- TO ALIGN THE STABLE MEMBER TO REFSSMAT
3
4 # CALLING SEQ -- CALL R51
5 # INPUT -- REFSMMAT
6 # OUTPUT -- GYRO TORQUE PULSES
7 # SUBROUTINES -- LOCSAM, PICAPAR, R52, R53, R54, R55
8
9 COUNT* $$/R51
10 R51 STQ
11 QMAJ
12 R51.1 EXIT
13 TC PHASCHNG
14 OCT 04024
15
16 R51C CAF OCT15
17 TC BANKCALL
18 CADR GOPERF1
19 TC GOTOPOOH
20 TC +2 # V33E
21 TC R51E # ENTER
22 TC INTERPRET
23 RTB DAD
24 LOADTIME
25 TSIGHT1
26 CALL
27 LOCSAM
28 EXIT
29 TC BANKCALL
30 CADR R56
31 R51F TC R51I
32 R51I TC R51E
33 TC ALARM
34 OCT 405
35 CAF VB05N09
36 TC BANKCALL
37 CADR GOFLASH
38 TC GOTOPOOH
39 TC R51E
40 TC R51C
41 R51E CAF ZERO
42 TS STARIND
43 R51.2 TC INTERPRET
44 R51.3 EXIT
45 TC PHASCHNG
46 OCT 04024
47
48 TC INTERPRET
49 CALL
50 R52 # AOP WILL MAKE CALLS TO SIGHTING
51 EXIT
52 TC BANKCALL
53
54
55
56
57
58
59
60

```

|    |           |        |           |    |
|----|-----------|--------|-----------|----|
| 1  | # R51 R55 |        | PAGE 212  | 1  |
| 2  |           | CADR   | AOTMARK   | 2  |
| 3  |           | TC     | BANKCALL  | 3  |
| 4  |           | CADR   | OPTSTALL  | 5  |
| 5  |           | TC     | CURTAINS  | 6  |
| 6  |           | CCS    | STARIND   | 7  |
| 7  |           | TCF    | +2        | 9  |
| 8  |           | TC     | R51.4     | 10 |
| 9  |           | TC     | INTPRET   | 11 |
| 10 |           | VLOAD  |           | 12 |
| 11 |           |        | STARAD +6 | 13 |
| 12 |           | STORE  | STARSAV2  | 14 |
| 13 |           | EXIT   |           | 15 |
| 14 |           | TC     | PHASCHNG  | 16 |
| 15 |           | OCT    | 04024     | 17 |
| 16 |           |        |           | 18 |
| 17 |           | TC     | INTPRET   | 19 |
| 18 |           | DLOAD  | CALL      | 20 |
| 19 |           |        | TSIGHT    | 21 |
| 20 |           |        | PLANET    | 22 |
| 21 |           | MXV    | UNIT      | 23 |
| 22 |           |        | REFSMMAT  | 24 |
| 23 |           | STOVL  | STARAD +6 | 25 |
| 24 |           |        | PLANVEC   | 26 |
| 25 |           | MXV    | UNIT      | 27 |
| 26 |           |        | REFSMMAT  | 28 |
| 27 |           | STOVL  | STARAD    | 29 |
| 28 |           |        | STARSAV1  | 30 |
| 29 |           | STOVL  | 6D        | 31 |
| 30 |           |        | STARSAV2  | 32 |
| 31 |           | STCALL | 12D       | 33 |
| 32 |           |        | R54       | 34 |
| 33 |           | BOFF   | CALL      | 35 |
| 34 |           |        | FREEFLAG  | 36 |
| 35 |           |        | R51K      | 37 |
| 36 |           |        | AXISGEN   | 38 |
| 37 |           | CALL   |           | 39 |
| 38 |           |        | R55       | 40 |
| 39 |           | CLEAR  |           | 41 |
| 40 |           |        | PFRATFLG  | 42 |
| 41 | R51K      | EXIT   |           | 43 |
| 42 | R51P63    | CAF    | OCT14     | 44 |
| 43 |           | TC     | BANKCALL  | 45 |
| 44 |           | CADR   | GOPERF1   | 46 |
| 45 |           | TC     | GOTOP00H  | 47 |
| 46 |           | TC     | R51C      | 48 |
| 47 |           | TC     | INTPRET   | 49 |
| 48 |           | GOTO   |           | 50 |
| 49 |           |        | QMAJ      | 51 |
| 50 | R51.4     | TC     | INTPRET   | 52 |
| 51 |           | VLOAD  |           | 53 |
| 52 |           |        |           | 54 |
| 53 |           |        |           | 55 |
| 54 |           |        |           | 56 |
| 55 |           |        |           | 57 |
| 56 |           |        |           | 58 |
| 57 |           |        |           | 59 |
| 58 |           |        |           | 60 |
| 59 |           |        |           | 61 |
| 60 |           |        |           | 62 |

|       |           |
|-------|-----------|
| STORE | STARAD +6 |
| DLOAD | STARSAV1  |
|       | CALL      |

STARAD +6  
STARSAV1

STORE  
SSP

PLANVEC

STARIND

1

GOTO

R51.3

TSIGHT1

2DEC

36000

## # 6 MIN TO MARKING

# P51-P53

# GYRO TORQUE COARSE ALGNMENT

GYCOARS

STQ

CALL  
QMAJ  
CALCGTA

CLEAR

CLEAR  
DRIFTFLG  
REFSMFLG

EXIT

CAF

TC

V16N20  
BANKCALL

# MONITOR GIMBALS

CADR

CA

TC

GODSPR  
R55CDR  
BANKCALL

CADR

TC

CADR

IMUPULSE  
BANKCALL  
IMUSTALL

TC

TC

OCT

CURTAINS  
PHASCHNG  
04024

TC

AXC,1

INTPRET  
AXC,2

XSMD  
REFSMMAT

CALL

# STORE DESIRED REFSMMAT

CLEAR

MATMOVE  
SET  
PFRATFLG  
REFSMFLG

CALL

NCOARSE

# SET DRIFT AND INITIALIZE 1/PIPADT

GOTO

V16N20

VN

R51K  
1620

```
1 # R55 GYRO TORQUE
2 # FUNCTION -- COMPUTE AND SEND GYRO PULSES
3 # CALLING SEQ -- CALL R55
4 # INPUT -- X,Y,ZDC -- REFSMMAT WRT PRESENT STABLE MEMBER
5 # OUTPUT -- GYRO PULSES
6 # SUBROUTINES -- CALCGTA, GOFLASH, GODSPR, IMUFINE, IMUPULSE, GOPERF1
```

```
9 COUNT* $$/R55
```

```
10 R55 STQ
```

```
11 QMIN
```

```
12 CALL
```

```
13 CALCGTA
```

```
14 PULSEM EXIT
```

```
15 R55.1 CAF V06N93
```

```
16 TC BANKCALL
```

```
17 CADR GOFLASH
```

```
18 TC GOTOP00H
```

```
19 TC R55.2
```

```
20 R55.2 TC R55RET
```

```
21 TC PHASCHNG
```

```
22 OCT 00214
```

```
23 CA R55CDR
```

```
24 TC BANKCALL
```

```
25 CADR IMUPULSE
```

```
26 TC BANKCALL
```

```
27 CADR IMUSTALL
```

```
28 TC CURTAINS
```

```
29 TC PHASCHNG
```

```
30 OCT 04024
```

```
32 R55RET TC INTPRET
```

```
33 GOTO
```

```
34 QMIN
```

```
35 V06N93 VN 0693
```

```
36 R55CDR ECADR OGC
```

```
37 R54 = CHKSDATA
```

```
39 # ROUTINE NAME -- CHKSDATA DATE -- JAN 9, 1967
```

```
40 # MOD NO -- 0 LOG SECTION -- P51-P53
```

```
41 # MODIFICATION BY -- LONSKE ASSEMBLY --
```

```
42 #
```

```
43 # FUNCTIONAL DESCRIPTION -- CHECKS THE VALIDITY OF A PAIR OF STAR SIGHTINGS. WHEN A PAIR OF STAR SIGHTINGS ARE MADE
```

```
44 # BY THE ASTRONAUT THIS ROUTINE OPERATES AND CHECKS THE OBSERVED SIGHTINGS AGAINST STORED STAR VECTORS IN THE
```

```
45 # COMPUTER TO INSURE A PROPER SIGHTING WAS MADE. THE FOLLOWING COMPUTATIONS ARE PERFORMED --
```

```
46 # OS1 = OBSERVED STAR 1 VECTOR
```

```
47 # OS2 = OBSERVED STAR 2 VECTOR
```

```
48 # SS1 = STORED STAR 1 VECTOR
```

```
49 # SS2 = STORED STAR 2 VECTOR
```

```
50 # A1 = ARCCOS(OS1 - OS2)
```

```
51 # A2 = ARCCOS(SS1 - SS2)
```

```
52 # A = ABS(2(A1 - A2))
```



```
1 # THE ANGULAR DIFFERENCE IS DISPLAYED FOR ASTRONAUT ACCEPTANCE.
2 #
3 #
4 # EXIT MODE -- 1. FREEFLAG SET IMPLIES ASTRONAUT WANTS TO PROCEED
5 # 2. FREEFLAG RESET IMPLIES ASTRONAUT WANTS TO RECYCLE
6 #
7 # OUTPUT -- 1. VERB 6,NOUN 3 -- DISPLAYS ANGULAR DIFFERENCE BETWEEN 2 SETS OF STARS.
8 # 2. STAR VECTORS FROM STAR CATALOG ARE LEFT IN 6D AND 12D.
9 #
10 # ERASABLE INITIALIZATION REQUIRED --
11 # 1. MARK VECTORS ARE STORED IN STARAD AND STARAD +6.
12 # 2. CATALOG VECTORS ARE STORED IN 6D AND 12D.
```

```
13 #
14 # DEBRIS --
```

```
15
16 COUNT* $$/R54
17 CHKSDATA STQ SET
18 QMIN
19 FREEFLAG
20 CHKSAB AXC,1 # SET X1 TO STORE EPHEMERIS DATA
21 STARAD
22
23 CHKSB VLOAD* DOT* # CAL. ANGLE THETA
24 0,1
25 6,1
26 SL1 ACOS
27 STORE THETA
28 BOFF # BRANCH TO CHKSD IF THIS IS 2ND PASS
29 INVERT
30 FREEFLAG
31 CHKSD
32 FREEFLAG # CLEAR FREEFLAG
33 AXC,1 DLOAD # SET X1 TO MARK ANGLES
34 6D
35 THETA
36 STORE 18D
37 GOTO
38 CHKSD DLOAD CHKSB # RETURN TO CAL. 2ND ANGLE
39 DSU
40 THETA
41 18D
42 ABS RTB # COMPUTE POS DIFF
43 SGNAGREE
44 STORE NORMTEM1
45 SET EXIT
46 FREEFLAG
47 CAF VB6N5
48 TC BANKCALL
49 CADR GOFLASH
50 TCF GOTOPOOH
51 TC CHKSDA # PROCEED
52 TC INTPRET
53 CLEAR GOTO
54 FREEFLAG
55 CHKSDA TC QMIN
56 INTPRET
```

|   |   |
|---|---|
| 1 | 2 |
|---|---|

|    |          |                   |                        |    |
|----|----------|-------------------|------------------------|----|
| 1  |          |                   |                        | 1  |
| 2  | GOTO     |                   |                        | 2  |
| 3  | COARSRET |                   |                        | 3  |
| 4  | MATMOVE  | VLOAD*            | # TRANSFER MATRIX      | 4  |
| 5  |          | 0,1               |                        | 5  |
| 6  |          | STORE 0,2         |                        | 6  |
| 7  | VLOAD*   |                   |                        | 7  |
| 8  |          | 6D,1              |                        | 8  |
| 9  |          | STORE 6D,2        |                        | 9  |
| 10 | VLOAD*   |                   |                        | 10 |
| 11 |          | 12D,1             |                        | 11 |
| 12 |          | STORE 12D,2       |                        | 12 |
| 13 | RVQ      |                   |                        | 13 |
| 14 | DEGREE1  | DEC 46            | # 1 DEG SCALED CDU/2   | 14 |
| 15 | DEG359   | DEC 16338         | # 359 DEG SCALED CDU/2 | 15 |
| 16 | RDCDUS   | INHINT            | # READ CDUS            | 16 |
| 17 |          | CA CDUX           |                        | 17 |
| 18 |          | INDEX FIXLOC      |                        | 18 |
| 19 | TS 1     |                   |                        | 19 |
| 20 |          | CA CDUY           |                        | 20 |
| 21 |          | INDEX FIXLOC      |                        | 21 |
| 22 | TS 2     |                   |                        | 22 |
| 23 |          | CA CDUZ           |                        | 23 |
| 24 |          | INDEX FIXLOC      |                        | 24 |
| 25 | TS 3     |                   |                        | 25 |
| 26 |          | RELINT            |                        | 26 |
| 27 |          | TC DANZIG         |                        | 27 |
| 28 |          | COUNT* \$\$/INFLT |                        | 28 |
| 29 |          |                   |                        | 29 |
| 30 |          |                   |                        | 30 |
| 31 |          |                   |                        | 31 |
| 32 |          |                   |                        | 32 |
| 33 |          |                   |                        | 33 |
| 34 |          |                   |                        | 34 |
| 35 |          |                   |                        | 35 |
| 36 |          |                   |                        | 36 |
| 37 |          |                   |                        | 37 |
| 38 |          |                   |                        | 38 |
| 39 |          |                   |                        | 39 |
| 40 |          |                   |                        | 40 |
| 41 |          |                   |                        | 41 |
| 42 |          |                   |                        | 42 |
| 43 |          |                   |                        | 43 |
| 44 |          |                   |                        | 44 |
| 45 |          |                   |                        | 45 |
| 46 |          |                   |                        | 46 |
| 47 |          |                   |                        | 47 |
| 48 |          |                   |                        | 48 |
| 49 |          |                   |                        | 49 |
| 50 |          |                   |                        | 50 |
| 51 |          |                   |                        | 51 |
| 52 |          |                   |                        | 52 |
| 53 |          |                   |                        | 53 |
| 54 |          |                   |                        | 54 |
| 55 |          |                   |                        | 55 |
| 56 |          |                   |                        | 56 |
| 57 |          |                   |                        | 57 |
| 58 |          |                   |                        | 58 |
| 59 |          |                   |                        | 59 |
| 60 |          |                   |                        | 60 |

# NAME -- P51 -- IMU ORIENTATION DETERMINATION

# MOD. NO. 1 23 JAN 67

LOG SECTION -- P51-P53

# MOD BY STURLAUGSON

ASSEMBLY SUNDANCE REV56

#

# FUNCTIONAL DESCRIPTION

# DETERMINES THE INERTIAL ORIENTATION OF THE IMU. THE PROGRAM IS SELECTED BY DSKY ENTRY. THE SIGHTING (AOTMARK) ROUTINE IS CALLED TO COLLECT AND PROCESS MARKED-STAR DATA. AOTMARK (R53) RETURNS THE STAR NUMBER AND THE STAR LOS VECTOR IN STARAD +6. TWO STARS ARE THUS SIGHTED. THE ANGLE BETWEEN THE TWO STARS IS THEN CHECKED AT

# CHKSDATA (R54). REFSMMAT IS THEN COMPUTED AT AXISGEN.

#

# CALLING SEQUENCE

# THE PROGRAM IS CALLED BY THE ASTRONAUT BY DSKY ENTRY.

#

# SUBROUTINES CALLED

# GOPERF3

# GOPERF1

# GODSPR

# IMUCOARS

# IMUFIN20

# AOTMARK (R53)

# CHKSDATA (R54)

# MKRELEAS

# AXISGEN

# MATMOVE

#

# ALARMS

# NONE.

#

# ERASABLE INITIALIZATION

# IMU ZERO FLAG SHOULD BE SET.

#

# OUTPUT

# REFSMMAT

# REFSMFLG

#

# DEBRIS

# WORK AREA

# STARAD

# STARIND

# BESTI

# BESTJ

COUNT\* \$\$/P51

P51

TC  
CADRBANKCALL  
IMUCHK# IS ISS ON - IF NOT, IMUCHK WILL SEND  
# ALARM CODE 210 AND EXIT VIA GOTOPOOH.CAF  
TCOCT15  
BANKCALLCADR  
TCGOPERF1  
GOTOPOOH# TERM.  
# V33TCF  
TC  
OCTP51B  
PHASCHNG  
04024CAF  
TS  
TSZERO  
THETAD  
THETAD +1

# ZERO THE GIMBALS

TS  
CAF  
TCTHETAD +2  
V06N22  
BANKCALLCADR  
CAF  
TCGODSPRET  
V41K  
BANKCALL

# NOW DISPLAY COARSE ALIGN VERB 41

CADR  
TC  
CALLGODSPRET  
INTPRET

COARSE

EXIT  
TC

PHASCHNG

OCT  
TCF04024  
P51 +2

P51B

TC  
OCT  
TCPHASCHNG  
00014  
INTPRET

CALL

SSP

NCOARSE  
SETPD

STARIND

# INDEX -- STAR 1 OR 2

0

0

P51C

EXIT  
TC  
OCT

PHASCHNG

04024

TC  
CADRBANKCALL  
AOTMARK

# R53

TC  
CADR  
TCBANKCALL  
AOTSTALL  
CURTAINSCCS  
TCF  
TCSTARIND  
P51D +1  
INTPRET

|    |                                            |       |                                          |    |
|----|--------------------------------------------|-------|------------------------------------------|----|
| 1  | # P51 +50                                  |       |                                          | 1  |
| 2  | VLOAD                                      |       |                                          | 2  |
| 3  | STARAD +6                                  |       |                                          | 3  |
| 4  | P51D                                       | STORE | STARSAV1                                 | 4  |
| 5  |                                            | EXIT  |                                          | 5  |
| 6  |                                            | TC    | PHASCHNG                                 | 6  |
| 7  |                                            | OCT   | 04024                                    | 7  |
| 8  |                                            |       |                                          | 8  |
| 9  | CCS STARIND                                |       |                                          | 9  |
| 10 | TCF P51E                                   |       |                                          | 10 |
| 11 | TC PHASCHNG                                |       |                                          | 11 |
| 12 | OCT 04024                                  |       |                                          | 12 |
| 13 |                                            |       |                                          | 13 |
| 14 | TC INTPRET                                 |       |                                          | 14 |
| 15 | DLOAD CALL                                 |       |                                          | 15 |
| 16 | TSIGHT                                     |       |                                          | 16 |
| 17 | PLANET                                     |       |                                          | 17 |
| 18 | STORE PLANVEC                              |       |                                          | 18 |
| 19 | EXIT                                       |       |                                          | 19 |
| 20 | CAF BIT1                                   |       |                                          | 20 |
| 21 | TS STARIND                                 |       |                                          | 21 |
| 22 | P51E                                       | TCF   | P51C +1 # DO SECOND STAR                 | 22 |
| 23 |                                            | TC    | PHASCHNG                                 | 23 |
| 24 |                                            | OCT   | 04024                                    | 24 |
| 25 |                                            |       |                                          |    |
| 26 | TC INTPRET                                 |       |                                          | 26 |
| 27 | DLOAD CALL                                 |       |                                          | 27 |
| 28 | TSIGHT                                     |       |                                          | 28 |
| 29 | PLANET                                     |       |                                          | 29 |
| 30 | STOVL 12D                                  |       |                                          | 30 |
| 31 | PLANVEC                                    |       |                                          | 31 |
| 32 | STOVL 6D                                   |       |                                          | 32 |
| 33 | STARSAV1                                   |       |                                          | 33 |
| 34 | STOVL STARAD                               |       |                                          | 34 |
| 35 | STARSAV2                                   |       |                                          | 35 |
| 36 | STCALL STARAD +6                           |       |                                          | 36 |
| 37 | CHKSDATA # CHECK STAR ANGLES IN STARAD AND |       |                                          | 37 |
| 38 | BON EXIT                                   |       |                                          | 38 |
| 39 | FREEFLAG                                   |       |                                          | 39 |
| 40 | P51G                                       |       |                                          | 40 |
| 41 | P51G                                       | TC    | P51 +2                                   | 41 |
| 42 |                                            | CALL  |                                          | 42 |
| 43 |                                            | AXC,1 | AXISGEN # COME BACK WITH REFSMMAT IN XDC | 43 |
| 44 |                                            | AXC,2 |                                          | 44 |
| 45 | XDC                                        |       |                                          | 45 |
| 46 | REFSMMAT                                   |       |                                          | 46 |
| 47 | CALL                                       |       |                                          | 47 |
| 48 | MATMOVE                                    |       |                                          | 48 |
| 49 | SET EXIT                                   |       |                                          | 49 |
| 50 | REFSMFLG                                   |       |                                          | 50 |
| 51 | TC GOTOP00H # FINIS                        |       |                                          | 51 |
| 52 |                                            |       |                                          | 52 |
| 53 |                                            |       |                                          | 53 |
| 54 |                                            |       |                                          | 54 |
| 55 |                                            |       |                                          | 55 |
| 56 |                                            |       |                                          | 56 |
| 57 |                                            |       |                                          | 57 |
| 58 |                                            |       |                                          | 58 |
| 59 |                                            |       |                                          | 59 |
| 60 |                                            |       |                                          | 60 |

|    |                |       |          |    |
|----|----------------|-------|----------|----|
| 1  |                |       |          | 1  |
| 2  | V41K<br>COARSE | VN    | 4100     | 2  |
| 3  |                | EXIT  |          | 3  |
| 4  |                | TC    | BANKCALL | 4  |
| 5  |                | CADR  | IMUCOARS | 5  |
| 6  |                | TC    | BANKCALL | 6  |
| 7  |                | CADR  | IMUSTALL | 7  |
| 8  |                | TC    | CURTAINS | 8  |
| 9  |                | TC    | BANKCALL | 9  |
| 10 |                | CADR  | IMUFINE  | 10 |
| 11 |                | TC    | BANKCALL | 11 |
| 12 |                | CADR  | IMUSTALL | 12 |
| 13 |                | TC    | CURTAINS | 13 |
| 14 |                | TC    | INTPRET  | 14 |
| 15 |                | RVQ   |          | 15 |
| 16 | NCOARSE        | EXIT  |          | 16 |
| 17 |                | CA    | TIME1    | 17 |
| 18 |                | TS    | 1/PIPADT | 18 |
| 19 |                | CS    | ZERO     | 19 |
| 20 |                | TS    | PIPAX    | 20 |
| 21 |                | TS    | PIPAY    | 21 |
| 22 |                | TS    | PIPAZ    | 22 |
| 23 |                | TC    | INTPRET  | 23 |
| 24 |                | VLOAD |          | 24 |
| 25 |                |       | ZEROVEC  | 25 |
| 26 |                | STORE | GCOMP    | 26 |
| 27 |                | SET   | RVQ      | 27 |
| 28 |                |       | DRIFTFLG | 28 |
| 29 |                |       |          | 29 |
| 30 |                |       |          | 30 |
| 31 |                |       |          | 31 |
| 32 |                |       |          | 32 |
| 33 |                |       |          | 33 |
| 34 |                |       |          | 34 |
| 35 |                |       |          | 35 |
| 36 |                |       |          | 36 |
| 37 |                |       |          | 37 |
| 38 |                |       |          | 38 |
| 39 |                |       |          | 39 |
| 40 |                |       |          | 40 |
| 41 |                |       |          | 41 |
| 42 |                |       |          | 42 |
| 43 |                |       |          | 43 |
| 44 |                |       |          | 44 |
| 45 |                |       |          | 45 |
| 46 |                |       |          | 46 |
| 47 |                |       |          | 47 |
| 48 |                |       |          | 48 |
| 49 |                |       |          | 49 |
| 50 |                |       |          | 50 |
| 51 |                |       |          | 51 |
| 52 |                |       |          | 52 |
| 53 |                |       |          | 53 |
| 54 |                |       |          | 54 |
| 55 |                |       |          | 55 |
| 56 |                |       |          | 56 |
| 57 |                |       |          | 57 |
| 58 |                |       |          | 58 |
| 59 |                |       |          | 59 |
| 60 |                |       |          | 60 |

# P51-P53

|    |                                                                        |        |            |    |
|----|------------------------------------------------------------------------|--------|------------|----|
| 1  |                                                                        |        |            | 1  |
| 2  | # NAME -- S52.2                                                        |        |            | 2  |
| 3  | # FUNCTION -- COMPUTE GIMBAL ANGLES FOR DESIRED SM AND PRESENT VEHICLE |        |            | 3  |
| 4  | # CALL -- CALL S52.2                                                   |        |            | 4  |
| 5  | # INPUT -- X,Y,ZSMD                                                    |        |            | 5  |
| 6  | # OUTPUT -- OGC,IGC,MGC,THETAD,+1,+2                                   |        |            | 6  |
| 7  | # SUBROUTINES -- CDUTRIG, CALCSMSC, MATMOVE, CALCGA                    |        |            | 7  |
| 8  |                                                                        |        |            | 8  |
| 9  |                                                                        | COUNT* | \$\$/S52.1 | 9  |
| 10 | S52.2                                                                  | STQ    | CALL       | 10 |
| 11 |                                                                        |        | QMAJ       | 11 |
| 12 |                                                                        |        | CDUTRIG    | 12 |
| 13 |                                                                        | CALL   |            | 13 |
| 14 |                                                                        |        | CALCSMSC   | 14 |
| 15 |                                                                        | AXT,1  | SSP        | 15 |
| 16 |                                                                        |        | 18D        | 16 |
| 17 |                                                                        |        | S1         | 17 |
| 18 |                                                                        |        | 6D         | 18 |
| 19 | S52.2A                                                                 | VLOAD* | VXM        | 19 |
| 20 |                                                                        |        | XNB +18D,1 | 20 |
| 21 |                                                                        |        | REFSMMAT   | 21 |
| 22 |                                                                        | UNIT   |            | 22 |
| 23 |                                                                        | STORE  | XNB +18D,1 | 23 |
| 24 |                                                                        | TIX,1  |            | 24 |
| 25 |                                                                        |        | S52.2A     | 25 |
| 26 | S52.2.1                                                                | AXC,1  | AXC,2      | 26 |
| 27 |                                                                        |        | XSMD       | 27 |
| 28 |                                                                        |        | XSM        | 28 |
| 29 |                                                                        | CALL   |            | 29 |
| 30 |                                                                        |        | MATMOVE    | 30 |
| 31 |                                                                        | CALL   |            | 31 |
| 32 |                                                                        |        | CALCGA     | 32 |
| 33 |                                                                        | GOTO   |            | 33 |
| 34 |                                                                        |        | QMAJ       | 34 |
| 35 |                                                                        |        |            | 35 |
| 36 |                                                                        |        |            | 36 |
| 37 |                                                                        |        |            | 37 |
| 38 |                                                                        |        |            | 38 |
| 39 |                                                                        |        |            | 39 |
| 40 |                                                                        |        |            | 40 |
| 41 |                                                                        |        |            | 41 |
| 42 |                                                                        |        |            | 42 |
| 43 |                                                                        |        |            | 43 |
| 44 |                                                                        |        |            | 44 |
| 45 |                                                                        |        |            | 45 |
| 46 |                                                                        |        |            | 46 |
| 47 |                                                                        |        |            | 47 |
| 48 |                                                                        |        |            | 48 |
| 49 |                                                                        |        |            | 49 |
| 50 |                                                                        |        |            | 50 |
| 51 |                                                                        |        |            | 51 |
| 52 |                                                                        |        |            | 52 |
| 53 |                                                                        |        |            | 53 |
| 54 |                                                                        |        |            | 54 |
| 55 |                                                                        |        |            | 55 |
| 56 |                                                                        |        |            | 56 |
| 57 |                                                                        |        |            | 57 |
| 58 |                                                                        |        |            | 58 |
| 59 |                                                                        |        |            | 59 |
| 60 |                                                                        |        |            | 60 |



# P51-P53

# NAME -- S52.3  
# FUNCTION -- XSMD= UNIT R  
# YSMD= UNIT(V X R)  
# ZSMD= UNIT(XSMD X YSMD)  
# CALL -- DLOAD CALL  
# TALIGN  
# S52.3  
# INPUT -- TIME OF ALIGNMENT IN MPAC  
# OUTPUT -- X,Y,ZSMD  
# SUBROUTINES -- CSMCONIC

|       |        |            |
|-------|--------|------------|
| S52.3 | COUNT* | \$\$/S52.3 |
|       | STQ    |            |
|       |        | QMAJ       |
|       | STCALL | TDEC1      |
|       |        | LEMCONIC   |
|       | VLOAD  | UNIT       |
|       |        | RATT       |
|       | STOVL  | XSMD       |
|       |        | VATT       |
|       | VXV    | UNIT       |
|       |        | RATT       |
|       | STOVL  | YSMD       |
|       |        | XSMD       |
|       | VXV    | UNIT       |
|       |        | YSMD       |
|       | STCALL | ZSMD       |
|       |        | QMAJ       |

# NAME -- R52 (AUTOMATIC OPTICS POSITIONING ROUTINE)

#

# FUNCTION -- POINT THE AOT OPTICS AXIS BY MANEUVERING THE LEM TO A NAVIGATION

# STAR SELECTED BY ALIGNMENT PROGRAMS OR DSKY INPUT

#

# CALLING -- CALL R52

#

# INPUT -- BESTI AND BESTJ (STAR CODES TIMES 6)

#

# OUTPUT -- STAR CODE IN BITS 1-6, DETENT CODE IN BITS 7-9

# (NO CHECK IS MADE TO INSURE THE DETENT CODE TO BE VALID)

# POINTVSM-1/2 UNIT NAV STAR VEC IN SM

# SCAXIS-AOT OPTIC AXIS VEC IN NB X-Z PLANE

#

# SUBROUT -- R60LEM

R52 COUNT\* \$\$/R52

STQ EXIT

SAVQR52

INDEX STARIND

CA BESTI # PICK UP STARCODE DETERMINED BY R56

EXTEND

MP 1/6TH

AD BIT8 # SET DETENT POSITION 2

TS STARCODE # SCALE AND STORE IN STARCODE

R52A CAF V01N70

TC BANKCALL

CADR GOFLASH # DISPLAY STARCODE AND WAIT FOR RESPONSE

TC GOTOP00H # V34 -- TERMINATE

TCF R52B # V33 -- PROCEED TO ORIENT LEM

TCF R52A # ENTER -- SELECT NEW STARCODE -- RECYCLE

R52B TC DOWNFLAG

ADRES 3AXISFLG # BIT6 OF FLAGWRD5 ZERO TO ALLOW VECPOINT

CA STARCODE # GRAB DETENT CODE

MASK HIGH9

EXTEND

MP BIT9

TS L # TEMP STORE DETENT

EXTEND

BZMF GETAZEL # CODE 0, COAS CALIBRATION

AD NEG7

EXTEND

BZF GETAZEL # CODE 7, COAS SIGHTING

EBANK= XYMARK

CA EBANK7

TS EBANK

|         |        |          |                                        |
|---------|--------|----------|----------------------------------------|
|         | INDEX  | L        |                                        |
|         | CA     | AOTAZ -1 | # PICK UP AZ CORRESPONDING TO DETENT   |
|         | TS     | L        |                                        |
|         | EBANK= | XSM      |                                        |
|         | CA     | EBANK5   | # CHANGE TO EBANK5 BUT DON'T DISTURB L |
|         | TS     | EBANK    |                                        |
|         | CA     | BIT13    | # SET ELV TO 45 DEG                    |
|         | XCH    | L        | # SET C(A)=AZ, C(L)=45 DEG             |
|         | TCF    | AZEL     | # GO COMP OPTIC AXIS                   |
| GETAZEL | CAF    | V06N87   | # CODE 0 OR 7, GET AZ AND EL KEY IN    |
|         | TC     | BANKCALL |                                        |
|         | CADR   | GOFLASH  |                                        |
|         | TC     | GOTOP00H | # V34 -- TERMINATE                     |
|         | TCF    | +2       | # PROCEED -- CALC OPTIC AXIS           |
|         | TCF    | GETAZEL  | # ENTER -- RECYCLE                     |
|         | EXTEND |          |                                        |
|         | DCA    | AZ       | # PICK UP AZ AND EL IN SP 2'S COMP     |
| AZEL    | INDEX  | FIXLOC   | # JAM AZ AND EL IN 8 AND 9 OF VAC      |
|         | DXCH   | 8D       |                                        |
|         | TC     | INTPRET  |                                        |
|         | CALL   |          | # GO COMPUTE OPTIC AXIS AND STORE IN   |
|         |        | OANB     | # SCAXIS IN NB COORDS                  |
|         | RTB    | CALL     |                                        |
|         |        | LOADTIME |                                        |
|         |        | PLANET   |                                        |
|         | MXV    | UNIT     |                                        |
|         |        | REFSMMAT |                                        |
|         | STORE  | POINTVSM | # STORE FOR VECPOINT                   |
|         | EXIT   |          |                                        |
|         | TC     | BANKCALL |                                        |
|         | CADR   | R60LEM   | # GO TORQUE LEM OPTIC AXIS TO STAR LOS |
|         | CAF    | HIGH9    | # IF COAS CALIBRATION CODE 0. RECYCLE  |
|         | MASK   | STARCODE |                                        |
|         | EXTEND |          |                                        |
|         | BZF    | R52A     |                                        |
|         | TC     | INTPRET  | # RETURN FROM KALCMANU                 |
|         | GOTO   |          |                                        |
|         |        | SAVQR52  | # RETURN TO CALLER                     |
| 1/6TH   | DEC    | .1666667 |                                        |
| V01N70  | VN     | 0170     |                                        |
| V06N87  | VN     | 687      |                                        |

## # LUNAR SURFACE STAR ACQUISITION

BANK 15  
SETLOC P50S  
BANK

COUNT\* \$\$/R59

R59

CS FLAGWRD3

MASK REFSMBIT # IF REFSMMAT FLAG CLEAR BYPASS STAR ACQUIRE

CCS A

TCF R59OUT # NO REFSMMAT GO TO AOTMARK

CAF V01N70\*  
TC BANKCALL

# SELECT STAR CODE FOR ACQUISITION

CADR GOFLASH  
TC GOTOP00H  
TCF R59A

# V34 -- TERMINATE

# V33 -- PROCEED

TCF R59

# V32 -- RECYCLE

R59A

CS HIGH9

# GRAB STARCODE FOR INDEX

MASK AOTCODE

EXTEND

MP REVCNT

# JUST 6

XCH L

INDEX STARIND

TS BESTI

INDEX FIXLOC

TS X1

# CODE X 6 FOR CATLOG STAR INDEX

EXTEND

BZF R59OUT

# BYPASS ACQUISITION IF NOT CATLOG STAR

COM

AD DEC227

EXTEND

BZMF R59OUT

TC INTPRET  
VLOAD\* MXV

CATLOG,1 # GRAB STAR VECTOR

# TRANSFORM TO SM

UNIT CALL  
CDU\*SMNB

STORE STAR

# TEMP STORE STAR VEC(NB)

EXIT

CAF BIT1  
TS POSCODE

# INITIALIZE AZ POSITION COD TO 1 (-60)

INCAZ

EBANK= XYMARK  
CA EBANK7  
TS EBANK

|        |          |                                       |
|--------|----------|---------------------------------------|
| INDEX  | POSCODE  |                                       |
| CA     | AOTAZ -1 | # PICK UP AZ CORRESPONDING TO POSCODE |
| TS     | L        |                                       |
| EBANK= | XSM      |                                       |
| CA     | EBANK5   |                                       |
| TS     | EBANK    |                                       |
| CA     | BIT13    | # SET ELV TO 45 DEG                   |
| XCH    | L        | # SET C(A)=AZ, C(L)=45 DEG            |
| TS     | QMIN     | # STORE QMIN=AZ FOR LATER             |
| INDEX  | FIXLOC   |                                       |
| DXCH   | 8D       | # JAM AZ IN 8D, 45 DEG IN 9D FOR OANB |
| TC     | INTPRET  |                                       |
| CALL   | OANB     | # GO CALC OPTIC AXIS WRT NB           |
| VLOAD  | DOT      |                                       |
|        | STAR     | # DOT STAR WITH OA                    |
|        | SCAXIS   |                                       |
| SL1    | ARCCOS   |                                       |
| STORE  | 24D      | # TEMP STORE ARCCOS(STAR.OPTAXIS)     |
| DSU    | BPL      |                                       |
|        | DEG30    | # SEE IF STAR IN AOT FIELD-OF-VIEW    |
|        | NXAX     | # NOT IN FIELD -- TRY NEXT POSITION   |
| DLOAD  | DSU      | # SEE IF STAR AT FIELD CENTER         |
|        | 24D      |                                       |
|        | DEG.5    |                                       |
| BMN    | DLOAD    | # CALC SPIRAL AND CURSOR              |
|        | ZSPCR    | # GO ZERO CURSOR AND SPIRAL           |
|        | 24D      | # GET SPIRAL                          |
| DMP    | SL4      |                                       |
|        | 3/4      | # 12 SCALED AT 16                     |
| STOVL  | 24D      | # 12(ARCCOS(AO.STAR)) SCALED IN REVS  |
| VXV    | SCAXIS   | # OA                                  |
|        | UNIT     |                                       |
| PUSH   | XUNIT    |                                       |
|        | VXV      | # OA X UNITX PD 0-5                   |
|        | SCAXIS   |                                       |
| VCOMP  |          |                                       |
| UNIT   | PDVL     | # UNIT(OA X (OA X UNITX)) PD 6-11     |
|        | SCAXIS   |                                       |
| VXV    | UNIT     |                                       |
|        | STAR     |                                       |
| PUSH   | DOT      | # 1/2(OA X STAR) PD 12-17             |
|        | 0        | # DOT WITH 1/2(OA X UNITX) FOR YROT   |
| SL1    | ARCCOS   |                                       |
| STOVL  | 26D      | # STORE THET SCALED IN REVS           |

|    |        |        |          |                                           |
|----|--------|--------|----------|-------------------------------------------|
| 1  |        |        |          |                                           |
| 2  |        | DOT    |          | # UP 12-17, UP 6-11 FOR C2                |
| 3  |        | BPL    | DLOAD    | # IF THET NEG -- GET 360-THET             |
| 4  |        |        | R59D     |                                           |
| 5  |        |        | ABOUTONE |                                           |
| 6  |        | DSU    |          |                                           |
| 7  |        |        | 26D      |                                           |
| 8  |        | STORE  | 26D      | # 360-THET SCALED IN REVS                 |
| 9  |        |        |          |                                           |
| 10 | R59D   | SLOAD  | SR1      |                                           |
| 11 |        |        | QMIN     | # RESCALE AZ(N) TO REVS                   |
| 12 |        | DAD    | PUSH     | # PUSH YROT + AZ(N) REVS                  |
| 13 |        |        | 26D      |                                           |
| 14 |        | RTB    |          |                                           |
| 15 |        |        | 1ST02S   |                                           |
| 16 |        | STODL  | CURSOR   | # YROT IN 1/2 REVS                        |
| 17 |        |        | 24D      | # LOAD SROT IN REVS                       |
| 18 |        | DAD    |          | # 12(SEP) + YROT                          |
| 19 |        | RTB    |          |                                           |
| 20 |        |        | 1ST02S   |                                           |
| 21 |        | STORE  | SPIRAL   | # SROT IN 1/2 REVS                        |
| 22 |        | EXIT   |          |                                           |
| 23 |        | TCF    | 79DISP   | # GO DISPLAY CURSOR-SPIRAL-POS CODE       |
| 24 |        |        |          |                                           |
| 25 | ZSPCR  | EXIT   |          |                                           |
| 26 |        | CAF    | ZERO     | # STAR ALMOST OPTIC AXIS, ZERO CURSOR     |
| 27 |        | TS     | CURSOR   | # AND SPIRAL ANGLES                       |
| 28 |        | TS     | SPIRAL   |                                           |
| 29 |        | TCF    | 79DISP   |                                           |
| 30 |        |        |          |                                           |
| 31 | NXAX   | EXIT   |          |                                           |
| 32 |        | INCR   | POSCODE  |                                           |
| 33 |        | CS     | POSCODE  |                                           |
| 34 |        | AD     | SEVEN    |                                           |
| 35 |        | EXTEND |          |                                           |
| 36 |        | BZMF   | R59ALM   | # THIS STAR NOT AT ANY POSITION           |
| 37 |        | TCF    | INCAZ    |                                           |
| 38 |        |        |          |                                           |
| 39 | R59ALM | TC     | ALARM    | # THIS STAR CAN'T BE LOCATED IN AOT FIELD |
| 40 |        | OCT    | 404      |                                           |
| 41 |        | CAF    | VB05N09  | # DISPLAY ALARM                           |
| 42 |        | TC     | BANKCALL |                                           |
| 43 |        | CADR   | GOFLASH  |                                           |
| 44 |        | TCF    | GOTOP00H | # VB34 -- TERMINATE                       |
| 45 |        | TCF    | R59OUT   | # VB33 -- PROCEED, GO WITHOUT ACQUIRE     |
| 46 |        | TCF    | R59      | # VB32 -- RECYCLE AND TRY ANOTHER STAR    |
| 47 |        |        |          |                                           |
| 48 | 79DISP | CAF    | V06N79   | # DISPLAY CURSOR, SPIRAL AND POS CODE     |
| 49 |        | TC     | BANKCALL |                                           |
| 50 |        | CADR   | GOFLASH  |                                           |
| 51 |        | TCF    | GOTOP00H | # V34 -- TERMINATE                        |
| 52 |        |        |          |                                           |
| 53 |        |        |          |                                           |
| 54 |        |        |          |                                           |
| 55 |        |        |          |                                           |
| 56 |        |        |          |                                           |
| 57 |        |        |          |                                           |
| 58 |        |        |          |                                           |
| 59 |        |        |          |                                           |
| 60 |        |        |          |                                           |

|    |         |        |            |                                            |    |
|----|---------|--------|------------|--------------------------------------------|----|
| 1  |         |        |            |                                            | 1  |
| 2  |         | TCF    | R59E       | # V33 -- PROCEED TO MARK ROUTINE           | 2  |
| 3  |         | TCF    | R59        | # V32 -- RECYCLE TO TOP OF R59 AGAIN       | 3  |
| 4  |         |        |            |                                            | 4  |
| 5  | R59E    | CAF    | SEVEN      | # GET DETENT CODE CORRESPONDING TO POSCODE | 5  |
| 6  |         | MASK   | POSCODE    |                                            | 6  |
| 7  |         | EXTEND |            |                                            | 7  |
| 8  |         | MP     | BIT7       | # DETEND CODE NOW IN L                     | 8  |
| 9  |         | CS     | HIGH9      |                                            | 9  |
| 10 |         | MASK   | AOTCODE    | # ISOLATE STAR NO BIT 1-6                  | 10 |
| 11 |         | AD     | L          |                                            | 11 |
| 12 |         | TS     | AOTCODE    | # STORE DETENT 7-9                         | 12 |
| 13 |         |        |            |                                            | 13 |
| 14 | R59OUT  | TC     | BANKCALL   | # GO TO AOTMARK FOR SIGHTING               | 14 |
| 15 |         | CADR   | AOTMARK    |                                            | 15 |
| 16 |         | TC     | BANKCALL   |                                            | 16 |
| 17 |         | CADR   | AOTSTALL   | # SLEEP TILL SIGHTING DONE                 | 17 |
| 18 |         | TC     | CURTAINS   | # BADEND RETURN FROM AOTMARK               | 18 |
| 19 |         | TCF    | R59RET     | # RETURN TO 1 STAR OR 2STAR                | 19 |
| 20 |         |        |            |                                            | 20 |
| 21 | V01N70* | VN     | 170        |                                            | 21 |
| 22 | V06N79  | VN     | 679        |                                            | 22 |
| 23 | DEG30   | 2DEC   | .083333333 | # 30 DEGREES                               | 23 |
| 24 | DEG.5   | 2DEC   | .00138888  | # .5 DEGREES SCALED IN REVS.               | 24 |
| 25 | DEG60   | OCT    | 12525      | # 60 DEG CDU SCALING                       | 25 |
| 26 | CURSOR  | EQUALS | GDT/2      |                                            | 26 |
| 27 | SPIRAL  | EQUALS | GDT/2 +2   |                                            | 27 |
| 28 | POSCODE | EQUALS | GDT/2 +4   |                                            | 28 |
| 29 |         |        |            |                                            | 29 |
| 30 |         |        |            |                                            | 30 |
| 31 |         |        |            |                                            | 31 |
| 32 |         |        |            |                                            | 32 |
| 33 |         |        |            |                                            | 33 |
| 34 |         |        |            |                                            | 34 |
| 35 |         |        |            |                                            | 35 |
| 36 |         |        |            |                                            | 36 |
| 37 |         |        |            |                                            | 37 |
| 38 |         |        |            |                                            | 38 |
| 39 |         |        |            |                                            | 39 |
| 40 |         |        |            |                                            | 40 |
| 41 |         |        |            |                                            | 41 |
| 42 |         |        |            |                                            | 42 |
| 43 |         |        |            |                                            | 43 |
| 44 |         |        |            |                                            | 44 |
| 45 |         |        |            |                                            | 45 |
| 46 |         |        |            |                                            | 46 |
| 47 |         |        |            |                                            | 47 |
| 48 |         |        |            |                                            | 48 |
| 49 |         |        |            |                                            | 49 |
| 50 |         |        |            |                                            | 50 |
| 51 |         |        |            |                                            | 51 |
| 52 |         |        |            |                                            | 52 |
| 53 |         |        |            |                                            | 53 |
| 54 |         |        |            |                                            | 54 |
| 55 |         |        |            |                                            | 55 |
| 56 |         |        |            |                                            | 56 |
| 57 |         |        |            |                                            | 57 |
| 58 |         |        |            |                                            | 58 |
| 59 |         |        |            |                                            | 59 |
| 60 |         |        |            |                                            | 60 |

|    |                  |                                                           |  |    |
|----|------------------|-----------------------------------------------------------|--|----|
| 1  |                  |                                                           |  | 1  |
| 2  | # NAME --        | PLANET                                                    |  | 2  |
| 3  | # FUNCTION --    | TO PROVIDE THE REFERENCE VECTOR FOR THE SIGHTED CELESTIAL |  | 3  |
| 4  | #                | BODY. STARS ARE FETCHED FROM THE CATALOG, SUN, EARTH AND  |  | 4  |
| 5  | #                | MOON ARE COMPUTED BY LOCSAM, PLANET VECTORS ARE ENTERED   |  | 5  |
| 6  | #                | BY DSKY INPUT.                                            |  | 6  |
| 7  | # CALL --        | CALL                                                      |  | 7  |
| 8  | #                | PLANET                                                    |  | 8  |
| 9  | # INPUT --       | TIME IN MPAC                                              |  | 9  |
| 10 | # OUTPUT --      | VECTOR IN MPAC                                            |  | 10 |
| 11 | # SUBROUTINES -- | LOCSAM                                                    |  | 11 |
| 12 | # DEBRIS --      | VAC, STARAD - STARAD +17                                  |  | 12 |
| 13 |                  |                                                           |  | 13 |
| 14 |                  | SETLOC P50S                                               |  | 14 |
| 15 |                  | BANK                                                      |  | 15 |
| 16 |                  | COUNT* \$\$/P51                                           |  | 16 |
| 17 |                  |                                                           |  | 17 |
| 18 | PLANET           | STOVL TSIGHT                                              |  | 18 |
| 19 |                  | ZEROVEC                                                   |  | 19 |
| 20 |                  | STORE STARAD                                              |  | 20 |
| 21 |                  | STQ EXIT                                                  |  | 21 |
| 22 |                  | GCTR                                                      |  | 22 |
| 23 |                  | CS HIGH9                                                  |  | 23 |
| 24 |                  | MASK AOTCODE                                              |  | 24 |
| 25 |                  |                                                           |  | 25 |
| 26 |                  | EXTEND                                                    |  | 26 |
| 27 |                  | MP REVCNT                                                 |  | 27 |
| 28 |                  | XCH L                                                     |  | 28 |
| 29 |                  | INDEX STARIND                                             |  | 29 |
| 30 |                  | TS BESTI                                                  |  | 30 |
| 31 |                  | CCS A                                                     |  | 31 |
| 32 |                  | TCF NOTPLAN                                               |  | 32 |
| 33 |                  | CAF VNPLANV                                               |  | 33 |
| 34 |                  | TC BANKCALL                                               |  | 34 |
| 35 |                  | CADR GOFLASH                                              |  | 35 |
| 36 |                  | TC -3                                                     |  | 36 |
| 37 |                  | TC +2                                                     |  | 37 |
| 38 |                  | TC -5                                                     |  | 38 |
| 39 |                  | TC INTPRET                                                |  | 39 |
| 40 |                  | VLOAD UNIT                                                |  | 40 |
| 41 |                  | STARAD                                                    |  | 41 |
| 42 |                  | GOTO                                                      |  | 42 |
| 43 |                  | GCTR                                                      |  | 43 |
| 44 | NOTPLAN          | CS A                                                      |  | 44 |
| 45 |                  | AD DEC227                                                 |  | 45 |
| 46 |                  | EXTEND                                                    |  | 46 |
| 47 |                  | BZMF CALSAM1                                              |  | 47 |
| 48 |                  | INDEX STARIND                                             |  | 48 |
| 49 |                  | CA BESTI                                                  |  | 49 |
| 50 |                  | INDEX FIXLOC                                              |  | 50 |
| 51 |                  | TS X1                                                     |  | 51 |
| 52 |                  | TC INTPRET                                                |  | 52 |
| 53 |                  |                                                           |  | 53 |
| 54 |                  |                                                           |  | 54 |
| 55 |                  |                                                           |  | 55 |
| 56 |                  |                                                           |  | 56 |
| 57 |                  |                                                           |  | 57 |
| 58 |                  |                                                           |  | 58 |
| 59 |                  |                                                           |  | 59 |
| 60 |                  |                                                           |  | 60 |



- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65
- 66
- 67
- 68
- 69
- 70
- 71
- 72
- 73
- 74
- 75
- 76
- 77
- 78
- 79
- 80

1412THE

1

```
1 # GRAVITY VECTOR DETERMINATION ROUTINE
2 # BY KEN VINCENT
3
4 #
5 # FOR DETAILED DESCRIPTION SEE 504GSOP 5.6.3.2.5.
6 #
7 # THIS PROGRAM FINDS THE DIRECTION OF THE MOON'S GRAVITY
8 # WHILE THE LM IS IN THE MOON'S SURFACE. IT WILL BE USED
9 # FOR LUNAR SURFACE ALIGNMENT. THE GRAVITY VECTOR IS
10 # DETERMINED BY READING THE PIPAS WITH THE IMU AT TWO
11 # PARTICULAR ORIENTATIONS. THE TWO READINGS ARE AVERAGED
12 # AND UNITIZED AND TRANSFORMED TO NB COORDINATES. THE TWO
13 # ORIENTATIONS WERE CHOSEN TO REDUCE BIAS ERRORS IN THE
14 # READINGS.
15 #
16 # CALL --
17 # TC BANKCALL
18 # CADR GVDETER
19 #
20 # INPUTS --
21 # PIPAS, CDUS
22 #
23 # OUTPUTS --
24 # STARSavl = UNIT GRAVITY
25 # GSAV = DITTO
26 # GRAVBIT = 1
27 #
28 # SUBROUTINES --
29 # PIPASR, IMUCOARS, IMUFINE, IMUSTALL, 1/PIPA, DELAYJOB, CDUTRIG,
30 # *NBSM*, *SNMB*, CALCGA, GOFLASH
31 #
32 # DEBRIS --
33 # VAC, SAC, STARAD, XSM, XNB, THETAD, DELV, COSCDU, SINCDU
34
35 GVDETER CAF 42DEG
36 TS THETAD
37
38 COM
39 TS THETAD +1
40 CAF 35DEG
41 TS THETAD +2
42 TC INTPRET
43 CLEAR CALL
44 REFSMFLG
45 LUNG
46
47 # FIND GIMBAL ANGLES WHICH ROTATE SM 180 DEG ABOUT G VEC
48 #
49 # DEFINE G COOR SYS
50 #
51 # [X̄] [UNIT G]
52 # * [] []
53 # M = [Ȳ] = [UNITEZSM * X̄]
54 # [] []
55 # [Z̄] [UNIT(X̄ * Ȳ)]
56 #
57 # THEN ROTATED SM WRT PRESENT IS
```

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} M = 2 \begin{pmatrix} X & X \\ I & J \end{pmatrix} - 1/2 I$$

ALSO NB WRT PRES SM IS

$$XNB = NBSM I$$

GIMBAL ANGLES = CALCGA( XSM, XNB )

SETLOC P50S  
BANK  
COUNT\* \$\$/P57

AXT,1 SSP # X1=18  
18D # S1=6  
S1 # X2, -2

LXC,2  
S1

GRAVEL VLOAD\* CALL  
XUNIT -6,2  
\*NBSM\* # SIN AND COS COMPUTED IN LUNG

STORE XNB +18D,1  
VLOAD

LXC,2 VXSC\* # COMPLEMENT -- UNITX ARE BACKWARD --  
X2  
STAR +6,2 # OUTER PRODUCT

VSL2 LXC,2  
X2  
VSU\* INCR,2

XUNIT -6,2  
2D  
STORE XSM +18D,1

TIX,1 CALL  
GRAVEL  
CALCGA

VLOAD VSR1  
GOUT  
STCALL STARAD +12D

LUNG  
VLOAD VSR1  
GOUT

VAD UNIT  
STARAD +12D  
STORE STARS1V1

DOT  
SL1 GSAV  
ACOS

|    |         |       |          |                                                 |
|----|---------|-------|----------|-------------------------------------------------|
| 1  |         | STORE | DSPTM1   |                                                 |
| 2  |         | EXIT  |          |                                                 |
| 3  |         | TC    | DOWNFLAG | # CLAR FREEFLAG IN CASE OF RECYCLE              |
| 4  |         | ADRES | FREEFLAG |                                                 |
| 5  |         |       |          |                                                 |
| 6  |         | CA    | DISGRVER |                                                 |
| 7  |         | TC    | BANKCALL |                                                 |
| 8  |         | CADR  | GOFLASH  |                                                 |
| 9  |         | TC    | GOTOPOOH |                                                 |
| 10 |         | TCF   | PROGRAV  | # VB33 -- PROCEED                               |
| 11 |         | TC    | UPFLAG   | # VB32 -- RECYCLE -- STORE GRAV AND DO IT AGAIN |
| 12 |         | ADRES | FREEFLAG | # AND SET FREEFLAG TO SHOW RECYCLE              |
| 13 |         |       |          |                                                 |
| 14 | PROGRAV | TC    | PHASCHNG |                                                 |
| 15 |         | OCT   | 04024    |                                                 |
| 16 |         |       |          |                                                 |
| 17 |         | TC    | INTPRET  |                                                 |
| 18 |         | VLOAD |          |                                                 |
| 19 |         |       | STARSAV1 |                                                 |
| 20 |         | STORE | GSAV     |                                                 |
| 21 |         | EXIT  |          |                                                 |
| 22 |         | CAF   | FREEFBIT | # IF FREEFLAG SET, RE-COMPUTE GRAVITY           |
| 23 |         | MASK  | FLAGWRDO |                                                 |
| 24 |         | CCS   | A        |                                                 |
| 25 |         | TCF   | GVDETER  | # SET                                           |
| 26 |         | TCF   | ATTCHK   | # EXIT FROM GVDETER                             |
| 27 |         |       |          |                                                 |
| 28 |         |       |          |                                                 |
| 29 | LUNG    | STQ   | VLOAD    |                                                 |
| 30 |         |       | QMIN     |                                                 |
| 31 |         |       | ZEROVEC  |                                                 |
| 32 |         | STORE | GACC     |                                                 |
| 33 |         | EXIT  |          |                                                 |
| 34 |         | TC    | PHASCHNG |                                                 |
| 35 |         | OCT   | 04024    |                                                 |
| 36 |         |       |          |                                                 |
| 37 |         | TC    | BANKCALL |                                                 |
| 38 |         | CADR  | IMUCOARS |                                                 |
| 39 |         | TC    | BANKCALL |                                                 |
| 40 |         | CADR  | IMUSTALL |                                                 |
| 41 |         | TC    | CURTAINS |                                                 |
| 42 |         | TC    | BANKCALL |                                                 |
| 43 |         | CADR  | IMUFINE  |                                                 |
| 44 |         | TC    | BANKCALL |                                                 |
| 45 |         | CADR  | IMUSTALL |                                                 |
| 46 |         | TC    | CURTAINS |                                                 |
| 47 |         | CA    | T/2SEC   |                                                 |
| 48 |         | TS    | GCTR     |                                                 |
| 49 |         | CA    | PRI031   |                                                 |
| 50 |         | TS    | 1/PIPADT |                                                 |
| 51 |         | TC    | BANKCALL |                                                 |
| 52 |         |       |          |                                                 |
| 53 |         |       |          |                                                 |
| 54 |         |       |          |                                                 |
| 55 |         |       |          |                                                 |
| 56 |         |       |          |                                                 |
| 57 |         |       |          |                                                 |
| 58 |         |       |          |                                                 |
| 59 |         |       |          |                                                 |
| 60 |         |       |          |                                                 |

|    |          |        |          |                                      |
|----|----------|--------|----------|--------------------------------------|
| 1  |          |        |          |                                      |
| 2  |          | CADR   | GCOMPZER | # INITIALIZE COMPENSATION            |
| 3  |          | TC     | PHASCHNG |                                      |
| 4  |          | OCT    | 04024    |                                      |
| 5  |          |        |          |                                      |
| 6  |          | TC     | BANKCALL | # DON'T NEED TO INHINT. THIS USED TO |
| 7  |          | CADR   | PIPSRINE | # INITIALIZE PIPAS. DON'T USE DATA   |
| 8  |          | TC     | INTPRET  |                                      |
| 9  | GREED    | EXIT   |          | # = MASK 7776 IN BASIC SO DON'T CARE |
| 10 |          | CAF    | 2SECS    |                                      |
| 11 |          | TC     | TWIDDLE  | # SET UP 2 SEC TASK TO READ PIPAS    |
| 12 |          | ADRES  | GRABGRAV |                                      |
| 13 |          |        |          |                                      |
| 14 |          | TC     | ENDOFJOB |                                      |
| 15 |          |        |          |                                      |
| 16 | GRABGRAV | TC     | IBNKCALL |                                      |
| 17 |          | CADR   | PIPSRINE |                                      |
| 18 |          | CAF    | PRI013   | # RE-ESTABLISH MAINLINE JOB          |
| 19 |          | TC     | FINDVAC  |                                      |
| 20 |          | EBANK= | STARAD   |                                      |
| 21 |          | 2CADR  | ADDGRAV  |                                      |
| 22 |          |        |          |                                      |
| 23 |          | TC     | TASKOVER |                                      |
| 24 |          |        |          |                                      |
| 25 | ADDGRAV  | TC     | BANKCALL |                                      |
| 26 |          | CADR   | 1/PIPA   |                                      |
| 27 |          | INCR   | GCTR     |                                      |
| 28 |          | TC     | INTPRET  |                                      |
| 29 |          | VLOAD  | VAD      |                                      |
| 30 |          |        | DELV     |                                      |
| 31 |          |        | GACC     |                                      |
| 32 |          | STORE  | GACC     | # ACCUMULATE G VECTOR                |
| 33 |          | SLOAD  | BMN      |                                      |
| 34 |          |        | GCTR     |                                      |
| 35 |          |        | GREED    |                                      |
| 36 |          | VLOAD  | UNIT     |                                      |
| 37 |          |        | GACC     |                                      |
| 38 |          | STCALL | STAR     |                                      |
| 39 |          |        | CDUTRIG  | # TRANSFORM IN NB COOR AND STORE     |
| 40 |          | CALL   |          | # IN OUTPUT                          |
| 41 |          |        | *SMNB*   |                                      |
| 42 |          | STORE  | GOUT     |                                      |
| 43 |          | EXIT   |          |                                      |
| 44 |          | TC     | PHASCHNG |                                      |
| 45 |          | OCT    | 04024    |                                      |
| 46 |          |        |          |                                      |
| 47 | QMINEXIT | TC     | INTPRET  |                                      |
| 48 |          | GOTO   |          |                                      |
| 49 |          |        | QMIN     |                                      |
| 50 | T/2SEC   | DEC    | -20      |                                      |
| 51 |          |        |          |                                      |
| 52 |          |        |          |                                      |
| 53 |          |        |          |                                      |
| 54 |          |        |          |                                      |
| 55 |          |        |          |                                      |
| 56 |          |        |          |                                      |
| 57 |          |        |          |                                      |
| 58 |          |        |          |                                      |
| 59 |          |        |          |                                      |
| 60 |          |        |          |                                      |

|          |     |       |
|----------|-----|-------|
| DISGRVER | VN  | 0604  |
| 42DEG    | OCT | 07357 |
| 35DEG    | OCT | 06211 |

# NAME -- GYROTRIM

#

# THIS PROGRAM COMPUTES AND SENDS GYRO COMMANDS WHICH CAUSE THE CDUS  
# TO ATTAIN A PRESCRIBED SET OF ANGLES. THIS ROUTINE ASSUMES THE  
# VEHICLES ATTITUDE REMAINS STATIONARY DURING ITS OPERATION.

#

# CALL CALL GYROTRIM

#

# INPUT THETAD,+1,+2 = DESIRED CDU ANGLES  
# CDUX,CDUY,CDUZ

#

# OUTPUT GYRO TORQUE PULSES

#

# SUBROUTINES TRG\*NBSM, \*NBSM\*, CDUTRIG, AXISGEN, CALCGTA, IMUFINE  
# IMPULSE, IMUSTALL

#

# DEBRIS CDUSPOT, SINC̄DU, COS̄CDU, STARAD, VAC, XDC, OḠC

COUNT\* \$\$/P57

GYROTRIM STQ DLOAD  
QMIN  
THETAD  
PDDL PDDL  
THETAD +2  
THETAD +1VDEF  
STOVL CDUSPOT  
XUNITCALL TRG\*NBSM  
STOVL STARAD  
YUNITCALL \*NBSM\*  
STCALL STARAD +6  
CDUTRIG

CALL CALCSMSC

VLOAD XNB

STOVL 6D  
YNB

STCALL 12D

CALL AXISGEN  
CALCGTAJUSTTRIM EXIT  
TC BANKCALL  
CADR IMUFINE  
TC BANKCALL

|      |          |
|------|----------|
| CADR | IMUSTALL |
| TC   | CURTAINS |

|      |          |
|------|----------|
| CA   | GYRCDR   |
| TC   | BANKCALL |
| CADR | IMUPULSE |

|      |          |
|------|----------|
| TC   | BANKCALL |
| CADR | IMUSTALL |

|     |          |
|-----|----------|
| TC  | CURTAINS |
| TCF | QMINEXIT |

| ECADR | OGC |
|-------|-----|
| 1     | 1   |
| 2     | 2   |
| 3     | 3   |
| 4     | 4   |
| 5     | 5   |
| 6     | 6   |
| 7     | 7   |
| 8     | 8   |
| 9     | 9   |
| 10    | 10  |
| 11    | 11  |
| 12    | 12  |
| 13    | 13  |
| 14    | 14  |
| 15    | 15  |
| 16    | 16  |
| 17    | 17  |
| 18    | 18  |
| 19    | 19  |
| 20    | 20  |
| 21    | 21  |
| 22    | 22  |
| 23    | 23  |
| 24    | 24  |
| 25    | 25  |
| 26    | 26  |
| 27    | 27  |
| 28    | 28  |
| 29    | 29  |
| 30    | 30  |
| 31    | 31  |
| 32    | 32  |
| 33    | 33  |
| 34    | 34  |
| 35    | 35  |
| 36    | 36  |
| 37    | 37  |
| 38    | 38  |
| 39    | 39  |
| 40    | 40  |
| 41    | 41  |
| 42    | 42  |
| 43    | 43  |
| 44    | 44  |
| 45    | 45  |
| 46    | 46  |
| 47    | 47  |
| 48    | 48  |
| 49    | 49  |
| 50    | 50  |
| 51    | 51  |
| 52    | 52  |
| 53    | 53  |
| 54    | 54  |
| 55    | 55  |
| 56    | 56  |
| 57    | 57  |
| 58    | 58  |
| 59    | 59  |
| 60    | 60  |
| 61    | 61  |
| 62    | 62  |
| 63    | 63  |
| 64    | 64  |
| 65    | 65  |
| 66    | 66  |
| 67    | 67  |
| 68    | 68  |
| 69    | 69  |
| 70    | 70  |
| 71    | 71  |
| 72    | 72  |
| 73    | 73  |
| 74    | 74  |
| 75    | 75  |
| 76    | 76  |
| 77    | 77  |
| 78    | 78  |
| 79    | 79  |
| 80    | 80  |
| 81    | 81  |
| 82    | 82  |
| 83    | 83  |
| 84    | 84  |
| 85    | 85  |
| 86    | 86  |
| 87    | 87  |
| 88    | 88  |
| 89    | 89  |
| 90    | 90  |
| 91    | 91  |
| 92    | 92  |
| 93    | 93  |
| 94    | 94  |
| 95    | 95  |
| 96    | 96  |
| 97    | 97  |
| 98    | 98  |
| 99    | 99  |
| 100   | 100 |

| GYRCDR | ECADR | OGC |
|--------|-------|-----|
|--------|-------|-----|



# PERFORM STAR ACQUISITION AND STAR SIGHTINGS

|        |        |           |                                       |
|--------|--------|-----------|---------------------------------------|
| 2STARS | CAF    | ZERO      | # INITIALIZE STARIND                  |
|        | TCF    | +2        | # ZERO FOR 1ST STAR, ONE FOR 2ND STAR |
| 1STAR  | CAF    | BIT1      |                                       |
|        | TS     | STARIND   |                                       |
|        | TC     | PHASCHNG  |                                       |
|        | OCT    | 04024     |                                       |
|        | TCF    | R59       | # GO DO STAR ACQUIRE AND AOTMARK      |
| R59RET | CA     | STARIND   | # BACK FROM SURFACE MARKING           |
|        | EXTEND |           |                                       |
|        | BZF    | ASTAR     | # 1ST STAR MARKED                     |
|        | TC     | PHASCHNG  |                                       |
|        | OCT    | 04024     |                                       |
|        | TC     | INTPRET   |                                       |
|        | DLOAD  | CALL      |                                       |
|        |        | TSIGHT    | # TIME OF 2ND MARK                    |
|        |        | PLANET    |                                       |
|        | STCALL | VEC2      | # STORE 2ND CATALOG VEC (REF)         |
|        |        | SURFLINE  |                                       |
| ASTAR  | TC     | INTPRET   |                                       |
|        | VLOAD  | STARAD +6 |                                       |
|        | STORE  | STARSAV1  | # 1ST OBSERVED STAR (SM)              |
|        | DLOAD  | CALL      |                                       |
|        |        | TSIGHT    | # TIME OF 1ST MARK                    |
|        |        | PLANET    |                                       |
|        | STORE  | VEC1      | # STORE 1ST CATALOG VEC (REF)         |
|        | EXIT   |           |                                       |
|        | TCF    | 1STAR     | # GO GET 2ND STAR SIGHTING            |

# DO FINE OR COARSE ALIGNMENT OF IMU

SURFLINE SSP AXT,2  
S2  
6

WRTDESIR

VLOAD\*

12D

MXV

VEC1 +12D,2

# PICK UP VEC IN REF, TRANS TO DESIRED SH

XSMD

UNIT

STORE

STARAD +12D,2

# VEC IN SM

VLOAD\*

STARSAV1 +12D,2

# PICK UP VEC IN PRESENT SM

STORE

18D,2

TIX,2

BON

WRTDESIR

INITALGN

# IF INITIAL PASS (OPTION 0) BYPASS R54

INITBY

DOALIGN

CALL

R54

# DO CHKSDATA

BOFF

FREEFLAG

P57POST

# ASTRO DOES NOT LIKE DATA TEST RESULTS

INITBY

CALL

AXISGEN

# GET DESIRED ORIENT WRT PRES. XDC,YDC,ZDC

CALL

CALCGTA

# GET GYRO TORQ ANGLES, OGC,IGC,MGC

EXIT

CAF

INITABIT

# IF INITIAL PASS BYPASS NOUN 93 DISPLAY

MASK

FLAGWRD8

CCS

A

TCF

5DEGTEST

CAF

DISPGYRO

# DISPLAY GYRO TORQ ANGLES V 06N93

TC

BANKCALL

CADR

GOFLASH

TC

GOTOPOOH

# V34 -- TERMINATE

TCF

5DEGTEST

# VB33 -- PROCEED TO COARSE OR FINE

TCF

P57POST +1

# VB32 -- RECYCLE, MAYBE RE-ALIGN

5DEGTEST

TC

INTPRET

# IF ANGLES GREATER THAN 5 DEGS, DO COARSE

VLOAD

BOV

OGC

SURFSUP

SURFSUP

STORE

OGCT

V/SC

BOV

5DEGREES

COATRIM

SSP

GOTO

QMIN

SURFDISP

JUSTTRIM # ANGLES LESS THAN 5 DEG, DO GYRO TORQ

SURFDISP

EXIT  
TC  
OCT  
PHASCHNG  
04024

TC  
AXC,1  
INTPRET  
AXC,2

SET  
REFSMFLG  
MATMOVE

EXIT

CCS  
TCF  
TCF  
OPTION2  
B2F8  
P57POST +1  
# IF OPTION ZERO DO FINISH

B2F8

CAF  
MASK  
INITABIT  
FLAGWRD8  
# IF INITIAL FLAG SET, RE-CYCLE.

CCS  
TCF  
TC  
A  
P57JUMP  
INTPRET  
# IT'S SET

CALL

REFMF  
# GO GET ATTITUDE VEC IN MF(YNBSAV,XNBSAV)

P57POST

EXIT

CAF  
TC  
CADR  
OCT14  
BANKCALL  
GOPERF1  
# DISPLAY V50N25 CHK CODE 14

TCF  
TCF  
CS  
GOTOPOOH  
P57JUMP  
BIT2  
# VB34 -- TERMINATE  
# VB33 -- PROCEED TO RE-ALIGN  
# TEST TO SEE IF ALIGNED BY OPTION 2

AD  
EXTEND  
OPTION2  
BZF  
+2  
# YES -- GO CALCULATE LANDING SITE

TCF  
TC  
OCT  
GOTOPOOH  
PHASCHNG  
04024  
# NO -- EXIT P57  
# RESTART PLACE

TC  
VLOAD  
INTPRET  
CALL  
GSAV  
# USE GNB

VXM  
CDU\*NBSM  
SET  
REFSMMAT  
# GO TO SM COORDS  
# ON MOON SO SET LUNAFLAG  
# G(REF) = (REFSMMAT)T (NBSM)GNB

PDVL  
LUNAFLAG  
ABVAL  
RLS

VXSC  
STORE  
CLEAR  
STADR  
ALPHAV  
RTB  
# ALPHAV = RLSMAG \* G(REF)



|    |          |          |                                           |    |
|----|----------|----------|-------------------------------------------|----|
| 1  |          |          |                                           | 1  |
| 2  |          |          |                                           | 2  |
| 3  | ERADFLAG |          |                                           | 3  |
| 4  | LOADTIME |          |                                           | 4  |
| 5  | CALL     |          |                                           | 5  |
| 6  |          | N89DISP  | # SUBROUTINE TO CALC LS AND GIVE RLS BACK | 6  |
| 7  | STORE    | RN       | # RN=RLS B-29 = LM POSITION               | 7  |
| 8  | VSL2     | PDDL     | # R-TO-RP GETS RLS B-27 AT 0-50 IN PDLIST | 8  |
| 9  |          | GDT/2 +4 | # TIME TEMP STORED IN N89DISP             | 9  |
| 10 | PUSH     |          | # TIME AT 6-7 IN PDLIST                   | 10 |
| 11 | STCALL   | PIPTIME  | # PIPTIME = LM STATE TIME                 | 11 |
| 12 |          | R-TO-RP  |                                           | 12 |
| 13 | STORE    | RLS      | # RLS IN MOON-FIXED COORDS                | 13 |
| 14 | EXIT     |          |                                           | 14 |
| 15 | TCF      | GOTOP00H | # EXIT P57                                | 15 |
| 16 |          |          |                                           | 16 |
| 17 |          |          |                                           | 17 |
| 18 |          |          |                                           | 18 |
| 19 |          |          |                                           | 19 |
| 20 |          |          |                                           | 20 |
| 21 |          |          |                                           | 21 |
| 22 |          |          |                                           | 22 |
| 23 |          |          |                                           | 23 |
| 24 |          |          |                                           | 24 |
| 25 |          |          |                                           | 25 |
| 26 |          |          |                                           | 26 |
| 27 |          |          |                                           | 27 |
| 28 |          |          |                                           | 28 |
| 29 |          |          |                                           | 29 |
| 30 |          |          |                                           | 30 |
| 31 |          |          |                                           | 31 |
| 32 |          |          |                                           | 32 |
| 33 |          |          |                                           | 33 |
| 34 |          |          |                                           | 34 |
| 35 |          |          |                                           | 35 |
| 36 |          |          |                                           | 36 |
| 37 |          |          |                                           | 37 |
| 38 |          |          |                                           | 38 |
| 39 |          |          |                                           | 39 |
| 40 |          |          |                                           | 40 |
| 41 |          |          |                                           | 41 |
| 42 |          |          |                                           | 42 |
| 43 |          |          |                                           | 43 |
| 44 |          |          |                                           | 44 |
| 45 |          |          |                                           | 45 |
| 46 |          |          |                                           | 46 |
| 47 |          |          |                                           | 47 |
| 48 |          |          |                                           | 48 |
| 49 |          |          |                                           | 49 |
| 50 |          |          |                                           | 50 |
| 51 |          |          |                                           | 51 |
| 52 |          |          |                                           | 52 |
| 53 |          |          |                                           | 53 |
| 54 |          |          |                                           | 54 |
| 55 |          |          |                                           | 55 |
| 56 |          |          |                                           | 56 |
| 57 |          |          |                                           | 57 |
| 58 |          |          |                                           | 58 |
| 59 |          |          |                                           | 59 |
| 60 |          |          |                                           | 60 |



|    |                             |       |       |    |
|----|-----------------------------|-------|-------|----|
| 1  | # COARSE AND FINE ALIGN IMU |       |       | 1  |
| 2  |                             |       |       | 2  |
| 3  |                             |       |       | 3  |
| 4  | COATRIM                     | AXC,1 | AXC,2 | 4  |
| 5  |                             |       | XDC   | 5  |
| 6  |                             |       | XSM   | 6  |
| 7  | CALL                        |       |       | 7  |
| 8  | CALL                        |       |       | 8  |
| 9  | CALL                        |       |       | 9  |
| 10 | CALL                        |       |       | 10 |
| 11 | CALL                        |       |       | 11 |
| 12 | CALL                        |       |       | 12 |
| 13 | CALL                        |       |       | 13 |
| 14 | BOFF                        |       |       | 14 |
| 15 | INITALGN                    |       |       | 15 |
| 16 | CORSIT                      |       |       | 16 |
| 17 | CAF                         |       |       | 17 |
| 18 | TC                          |       |       | 18 |
| 19 | CADR                        |       |       | 19 |
| 20 | TC                          |       |       | 20 |
| 21 | TCF                         |       |       | 21 |
| 22 | TCF                         |       |       | 22 |
| 23 | TC                          |       |       | 23 |
| 24 | OCT                         |       |       | 24 |
| 25 | TC                          |       |       | 25 |
| 26 | CALL                        |       |       | 26 |
| 27 | CALL                        |       |       | 27 |
| 28 | CALL                        |       |       | 28 |
| 29 | CALL                        |       |       | 29 |
| 30 | GOTO                        |       |       | 30 |
| 31 | VN                          |       |       | 31 |
| 32 |                             |       |       | 32 |
| 33 |                             |       |       | 33 |
| 34 |                             |       |       | 34 |
| 35 |                             |       |       | 35 |
| 36 |                             |       |       | 36 |
| 37 |                             |       |       | 37 |
| 38 |                             |       |       | 38 |
| 39 |                             |       |       | 39 |
| 40 |                             |       |       | 40 |
| 41 |                             |       |       | 41 |
| 42 |                             |       |       | 42 |
| 43 |                             |       |       | 43 |
| 44 |                             |       |       | 44 |
| 45 |                             |       |       | 45 |
| 46 |                             |       |       | 46 |
| 47 |                             |       |       | 47 |
| 48 |                             |       |       | 48 |
| 49 |                             |       |       | 49 |
| 50 |                             |       |       | 50 |
| 51 |                             |       |       | 51 |
| 52 |                             |       |       | 52 |
| 53 |                             |       |       | 53 |
| 54 |                             |       |       | 54 |
| 55 |                             |       |       | 55 |
| 56 |                             |       |       | 56 |
| 57 |                             |       |       | 57 |
| 58 |                             |       |       | 58 |
| 59 |                             |       |       | 59 |
| 60 |                             |       |       | 60 |

## # LUNAR SURFACE IMU ALIGNMENT PROGRAM

P57 TC BANKCALL # IS ISS ON -- IF NOT, IMUCHK WILL SEND  
CADR IMUCHK # ALARM CODE 210 AND EXIT VIA GOTOP00H

P57OPT CAF THREE # JAM REFSMMAT OPTION 3 FOR INITIAL DISP.  
TS OPTION2  
CAF BIT1

TC BANKCALL  
CADR GOPERF4R # FLASH V04N06 FOR ALIGNMENT CODE  
TC GOTOP00H # V34 TERMINATE

TCF ALIGNOPT # V33 PROCEED  
TCF P57OPT # V32 RECYCLE

TC PHASCHNG  
OCT 00014  
TC ENDOFJOB

ALIGNOPT CA OPTION2  
MASK THREE

INDEX A  
TCF +1  
TCF TDISP # OPTION 4 LS ORIENTATION

TCF PACKOPTN # OPTION 1 PREFERRED  
TCF P57OPT # OPTION 2 INVALID IN P57, RECYCLE  
TC INTPRET # OPTION 3 REFSMMAT

AXC,1 AXC,2 # JAM REFSMMAT IN XSMD LOC  
REFSMMAT  
XSMD

CALL  
GOTO MATMOVE

PACKOPTN -1

TDISP TC INTPRET

DLOAD  
P57A STORE TIG # LOAD ASCENT TIME FOR DISPLAY  
DSPTM1

P57AA EXIT  
CAF V06N34\* # DISPLAY TALIGN, TALIGN : DSPTM1  
TC BANKCALL

CADR GOFLASH  
TCF GOTOP00H # V34 -- TERMINATE  
TCF +2

TCF P57AA # VB32 -- RECYCLE

TC INTPRET  
RTB PDDL  
LOADTIME # PUSH CURRENT TIME AND PICK UP KEY IN  
DSPTM1

|    |          |        |            |                                               |
|----|----------|--------|------------|-----------------------------------------------|
| 1  |          |        |            |                                               |
| 2  |          | BZE    | PDDL       |                                               |
| 3  |          |        | P57C       | # IF KEY IN TIME ZERO - TALIGN=CURRENT TIME   |
| 4  |          | DSU    | BPL        | # NOT ZERO SO EXCHANGE PD WITH DSPTEM1        |
| 5  |          |        | DSPTEM1    |                                               |
| 6  |          |        | P57C       |                                               |
| 7  |          | DLOAD  | STADR      | # IF KEYIN TIME GREATER THAN CURRENT TIME     |
| 8  |          | STORE  | TIG        | # STORE IT IN TIG                             |
| 9  |          | STCALL | TALIGN     |                                               |
| 10 |          |        | P57D       |                                               |
| 11 | P57C     | DLOAD  | STADR      |                                               |
| 12 |          | STORE  | TALIGN     |                                               |
| 13 | P57D     | STCALL | TDEC1      |                                               |
| 14 |          |        | LEMPREC    | # COMPUTE DESIRED IMU ORIENTATION STORE       |
| 15 |          | VLOAD  | UNIT       | # IN X,Y,ZSMD                                 |
| 16 |          |        | RATT       |                                               |
| 17 |          | STCALL | XSMD       |                                               |
| 18 |          |        | LSORIENT   |                                               |
| 19 |          | EXIT   |            |                                               |
| 20 | PACKOPTN | CAF    | ZERO       | # PACK FLAG BITS FOR OPTION DISPLAY           |
| 21 |          | TS     | OPTION1 +1 | # JAM ZERO IN ALIGNMENT OPTION                |
| 22 |          | TS     | OPTION1 +2 | # INITIALIZE FLAG BIT CONFIGURATION           |
| 23 |          | CAF    | REFSMBIT   |                                               |
| 24 |          | MASK   | FLAGWRD3   | # REFSMFLG                                    |
| 25 |          | CCS    | A          |                                               |
| 26 |          | CAF    | BIT7       | # SET                                         |
| 27 |          | ADS    | OPTION1 +2 | # CLEAR -- JUST ZERO                          |
| 28 |          | CAF    | ATTFLBIT   |                                               |
| 29 |          | MASK   | FLAGWRD6   | # ATTFLG                                      |
| 30 |          | CCS    | A          |                                               |
| 31 |          | CAF    | BIT4       | # SET                                         |
| 32 |          | ADS    | OPTION1 +2 | # CLEAR -- ZERO IN A                          |
| 33 |          | CAF    | BIT4       |                                               |
| 34 |          | TS     | OPTION1    | # JAM 00010 IN OPTION1 FOR CHECK LIST         |
| 35 |          |        |            |                                               |
| 36 | DSPOPTN  | CAF    | VB05N06    | # DISPLAY OPTION CODE AND FLAG BITS           |
| 37 |          | TC     | BANKCALL   |                                               |
| 38 |          | CADR   | GOFLASH    |                                               |
| 39 |          | TCF    | GOTOP00H   | # VB34 -- TERMINATE                           |
| 40 |          | TCF    | +2         | # V33 -- PROCEED                              |
| 41 |          | TCF    | DSPOPTN    | # V32 -- RECYCLE                              |
| 42 |          |        |            |                                               |
| 43 |          | CAF    | REFSMBIT   |                                               |
| 44 |          | MASK   | FLAGWRD3   |                                               |
| 45 |          | CCS    | A          |                                               |
| 46 |          | TCF    | GETLMATT   | # SET, GO COMPUTE LM ATTITUDE                 |
| 47 |          | CAF    | ATTFLBIT   | # CLEAR -- CHECK ATTFLAG FOR STORED ATTITUDE. |
| 48 |          | MASK   | FLAGWRD6   |                                               |
| 49 |          | CCS    | A          |                                               |
| 50 |          | TCF    | BYLMATT    | # ALLFLG SET, CHK OPTION FOR GRAVITY COMP     |
| 51 |          | CAF    | BIT2       | # SEE IF OPTION 2 OR 3                        |
| 52 |          |        |            |                                               |
| 53 |          |        |            |                                               |
| 54 |          |        |            |                                               |
| 55 |          |        |            |                                               |
| 56 |          |        |            |                                               |
| 57 |          |        |            |                                               |
| 58 |          |        |            |                                               |
| 59 |          |        |            |                                               |
| 60 |          |        |            |                                               |

[illegible]





|    |                                                                  |        |          |    |
|----|------------------------------------------------------------------|--------|----------|----|
| 1  | # TRANSFORM VEC1,2 FROM MOON FIXED TO REF AND JAM BACK IN VEC1,2 |        |          | 1  |
| 2  |                                                                  |        |          | 2  |
| 3  |                                                                  |        |          | 3  |
| 4  | MFREF                                                            | STQ    | SETPD    | 4  |
| 5  |                                                                  |        | QMAJ     | 5  |
| 6  |                                                                  |        | 0        | 6  |
| 7  |                                                                  | RTB    |          | 7  |
| 8  |                                                                  |        | LOADTIME | 8  |
| 9  |                                                                  | STOVL  | TSIGHT   | 9  |
| 10 |                                                                  |        | VEC1     | 10 |
| 11 |                                                                  | PDDL   | PUSH     | 11 |
| 12 |                                                                  |        | TSIGHT   | 12 |
| 13 |                                                                  | CALL   |          | 13 |
| 14 |                                                                  |        | RP-TO-R  | 14 |
| 15 |                                                                  | STOVL  | VEC1     | 15 |
| 16 |                                                                  |        | VEC2     | 16 |
| 17 |                                                                  | SETPD  | PDDL     | 17 |
| 18 |                                                                  |        | 0        | 18 |
| 19 |                                                                  |        | TSIGHT   | 19 |
| 20 |                                                                  | PUSH   | CALL     | 20 |
| 21 |                                                                  |        | RP-TO-R  | 21 |
| 22 |                                                                  | STCALL | VEC2     | 22 |
| 23 |                                                                  |        | QMAJ     | 23 |
| 24 |                                                                  |        |          | 24 |
| 25 |                                                                  |        |          | 25 |
| 26 |                                                                  |        |          | 26 |
| 27 |                                                                  |        |          | 27 |
| 28 |                                                                  |        |          | 28 |
| 29 |                                                                  |        |          | 29 |
| 30 |                                                                  |        |          | 30 |
| 31 |                                                                  |        |          | 31 |
| 32 |                                                                  |        |          | 32 |
| 33 |                                                                  |        |          | 33 |
| 34 |                                                                  |        |          | 34 |
| 35 |                                                                  |        |          | 35 |
| 36 |                                                                  |        |          | 36 |
| 37 |                                                                  |        |          | 37 |
| 38 |                                                                  |        |          | 38 |
| 39 |                                                                  |        |          | 39 |
| 40 |                                                                  |        |          | 40 |
| 41 |                                                                  |        |          | 41 |
| 42 |                                                                  |        |          | 42 |
| 43 |                                                                  |        |          | 43 |
| 44 |                                                                  |        |          | 44 |
| 45 |                                                                  |        |          | 45 |
| 46 |                                                                  |        |          | 46 |
| 47 |                                                                  |        |          | 47 |
| 48 |                                                                  |        |          | 48 |
| 49 |                                                                  |        |          | 49 |
| 50 |                                                                  |        |          | 50 |
| 51 |                                                                  |        |          | 51 |
| 52 |                                                                  |        |          | 52 |
| 53 |                                                                  |        |          | 53 |
| 54 |                                                                  |        |          | 54 |
| 55 |                                                                  |        |          | 55 |
| 56 |                                                                  |        |          | 56 |
| 57 |                                                                  |        |          | 57 |
| 58 |                                                                  |        |          | 58 |
| 59 |                                                                  |        |          | 59 |
| 60 |                                                                  |        |          | 60 |

# COMPUTE LM ATTITUDE IN MOON FIXED COORDINATES USING REFSMMAT AND  
# STORE IN YNBSAV AND ZNBSAV.

|       |                |                              |                           |
|-------|----------------|------------------------------|---------------------------|
| REFMF | STQ            | CALL<br>QMAJ                 |                           |
|       | RTB            | CDUTRIG<br>SETPD<br>LOADTIME | # GET SIN AND COS OF CDUS |
|       | STCALL         | 0<br>TSIGHT<br>CALCSMSC      | # GET YNB IN SM           |
|       | VLOAD          | VXM<br>YNB<br>REFSMMAT       | # YNB TO REF              |
|       | UNIT           | PDDL<br>TSIGHT               |                           |
|       | PUSH           | CALL                         |                           |
|       | STOVL          | R-TO-RP<br>YNBSAV<br>ZNB     | # YNB TO MF               |
|       | VXM            | UNIT<br>REFSMMAT             | # ZNB TO REF              |
|       | PDDL           | PUSH                         |                           |
|       | CALL           | TSIGHT                       |                           |
|       | STORE<br>SETGO | R-TO-RP<br>ZNBSAV            | # ZNB TO MF               |
|       |                | ATTFLAG<br>QMAJ              |                           |

# BRANCH TO ALIGNMENT OPTION

|          |                      |                              |                                        |
|----------|----------------------|------------------------------|----------------------------------------|
| GETLMATT | TC<br>CALL           | INTPRET<br><br>REFMF         | # GO TRANSFORM TO MF IN YNBSAV, ZNBSAV |
|          | EXIT                 |                              |                                        |
| BYLMATT  | TC                   | UPFLAG                       | # SET INITIAL ALIGN FLAG               |
|          | ADRES<br>CAF<br>MASK | INITALGN<br>BIT1<br>OPTION2  | # SEE IF OPTION 1 OR 3                 |
|          | CCS<br>TCF           | A<br>GVDETER                 | # OPTION 1 OR 2, GET GRAVITY           |
| ATTCHK   | TC<br>OCT            | PHASCHNG<br>04024            |                                        |
|          | CAF<br>MASK<br>CCS   | ATTFLBIT<br>FLAGWRD6<br>A    | # NOT 1 OR 3, CHECK ATTFLAG            |
| P57JUMP  | TCF<br>TC<br>OCT     | P57OPT0<br>PHASCHNG<br>04024 | # GET ALIGNMENT VECs FOR OPTION 0      |
|          | TC<br>ADRES          | DOWNFLAG<br>INITALGN         | # ATTFLG CLEAR -- RESET INTALIGN FLAG  |
|          | CAF<br>MASK<br>INDEX | THREE<br>OPTION2<br>A        | # BRANCH ON OPTION CODE                |
|          | TCF                  | +1                           |                                        |
|          | TCF                  | P57OPT0                      | # OPTION IS 0                          |
|          | TCF                  | P57OPT1                      | # OPTION IS 1                          |
|          | TCF                  | P57OPT2                      | # OPTION IS 2                          |
|          | TCF                  | P57OPT3                      | # OPTION IS 3                          |

# OPTION 0, GET TWO ATTITUDE VECs

|         |        |          |                                       |
|---------|--------|----------|---------------------------------------|
| P57OPT0 | TC     | INTPRET  |                                       |
|         | VLOAD  | YNBSAV   | # Y AND Z ATTITUDE WILL BE PUT IN REF |
|         | STOVL  | VEC1     |                                       |
|         |        | ZNBSAV   |                                       |
|         | STCALL | VEC2     |                                       |
|         |        | CDUTRIG  |                                       |
|         | CALL   | CALCSMSC | # COMPUTE SC AXIS WRT PRESENT SM      |
|         | VLOAD  | YNB      |                                       |
| SAMETYP | STOVL  | STARSAV1 | # Y SC AXIS WRT PRESENT SM            |
|         |        | ZNB      |                                       |
|         | STCALL | STARSAV2 | # Z SC AXIS WRT PRESENT SM            |
|         |        | MFREF    | # TRANSFORM VEC1,2 FROM MF TO REF     |
|         | GOTO   | SURFLINE |                                       |

# OPTION 1, GET LANDING SITE AND Z-ATTITUDE VEC

|         |        |          |                                         |
|---------|--------|----------|-----------------------------------------|
| P57OPT1 | TC     | INTPRET  |                                         |
|         | VLOAD  | UNIT     |                                         |
|         |        | RLS      | # LANDING SITE VEC                      |
|         | STOVL  | VEC1     |                                         |
|         |        | ZNBSAV   | # Z ATTITUDE VEC                        |
|         | STCALL | VEC2     |                                         |
|         |        | CDUTRIG  |                                         |
|         | CALL   | CALCSMSC | # GET ZNB AXIS WRT PRES SM FOR STARSAV2 |
|         | VLOAD  | CALL     |                                         |
|         |        | GSAV     | # TRANS GSAV FROM NB TO SM FOR STARSAV1 |
|         |        | CDU*NBSM |                                         |
|         | GOTO   | SAMETYP  | # NOW DO SAME AS OPTION 0               |



# P51-P53

# OPTION 2, GET TWO STAR SIGHTINGS

P57OPT2            TCF        2STARS            # DO SIGHTING ON 2 STARS

# OPTION 3, GET LANDING SITE VEC AND ONE STAR SIGHTING

P57OPT3            TC        INTPRET  
                     VLOAD     UNIT

                     RLS            # LANDING SITE VEC

                     STORE     VEC1  
                     STOVL    VEC2

# DUMMY VEC2 FOR 2ND CATALOG STAR

                     GSAV            # GRAVITY VEC NB

                     CALL

                     CDU\*NBSM        # TRANS GSAV FROM NB TO SM FOR STARS

                     STCALL    STARS

# STARS IS STORED AS 2ND OBSERVED STAR

                     EXIT

                     TCF        1STAR

# 1STAR GET VEC2, STARS GOES TO SURFLINE.

VB05N06            VN        506

# CHECK IMODES30 TO VERIFY IMU IS ON

IMUCHK

CS

IMODES30

MASK

BIT9

CCS

A

# IS IMU ON

TCF

+4

# YES

TC

ALARM

# NO, SEND ALARM AND EXIT

OCT

210

TC

GOTOP00H

TC

UPFLAG

ADRES

IMUSE

# SET IMUSE FLAG

TC

SWRETURN

BANK

04

SETLOC

AOTMARK2

BANK

COUNT\*

\$\$/P57

LSORIENT

STQ

VLOAD

QMAJ

VXV

RRECTCSM

VXV

VRECTCSM

XSMD

UNIT

STORE

ZSMD

VXV

UNIT

XSMD

STCALL

YSMD

QMAJ

```
1 # NAME - LSPOS - LOCATE SUN AND MOON DATE - 25 OCT 67
2 # MOD NO.1
3 # MOD BY NEVILLE ASSEMBLY SUNDANCE
4 #
5 # FUNCTIONAL DESCRIPTION
6 #
7 # COMPUTES UNIT POSITION VECTOR OF THE SUN AND MOON IN THE BASIC REFERENCE SYSTEM. THE SUN VECTOR S IS
8 # LOCATED VIA TWO ANGLES. THE FIRST ANGLE(OBLIQUITY) IS THE ANGLE BETWEEN THE EARTH EQUATOR AND THE ECLIPTIC. THE
9 # SECOND ANGLE IS THE LONGITUDE OF THE SUN MEASURED IN THE ECLIPTIC.
10 # THE POSITION VECTOR OF THE SUN IS
11 #
12 # -
13 # S=(COS(LOS), COS(OBL)*SIN(LOS), SIN(OBL)*SIN(LOS)), WHERE
14 #
15 # LOS=LOS +LOS *T-(C *SIN(2PI*T)/365.24 +C *COS(2PI*T)/365.24)
16 # 0 R 0 1
17 # LOS (RAD) IS THE LONGITUDE OF THE SUN FOR MIGNIGHT JUNE 30TH OF THE PARTICULAR YEAR.
18 # 0
19 # LOS (RAD/DAY) IS THE MEAN RATE FOR THE PARTICULAR YEAR.
20 # R
21 # LOS AND LOS ARE STORED AS LOSO AND LOSR IN RATESP.
22 # 0 R
23 # COS(OBL) AND SIN(OBL) ARE STORED IN THE MATRIX KONMAT.
24 # T, TIME MEASURED IN DAYS(24 HOURS), IS STORED IN TIMEP.
25 # C AND C ARE FUDGE FACTORS TO MINIMIZE THE DEVIATION. THEY ARE STORED AS ONE CONSTANT(CMOD), SINCE
26 # 0 1 2 2 1/2
27 # C *SIN(X)+C *COS(X) CAN BE WRITTEN AS (C +C) *SIN(X+PHI), WHERE PHI=ARCTAN(C /C).
28 # 0 1 0 1 1 0
29 #
30 # THE MOON IS LOCATED VIA FOUR ANGLES. THE FIRST IS THE OBLIQUITY. THE SECOND IS THE MEAN LONGITUDE OF THE MOON,
31 # MEASURED IN THE ECLIPTIC FROM THE MEAN EQUINOX TO THE MEAN ASCENDING NODE OF THE LUNAR ORBIT, AND THEN ALONG THE
32 # ORBIT. THE THIRD ANGLE IS THE ANGLE BETWEEN THE ECLIPTIC AND THE LUNAR ORBIT. THE FOURTH ANGLE IS THE LONGITUDE
33 # OF THE NODE OF THE MOON, MEASURED IN THE LUNAR ORBIT. LET THESE ANGLES BE OBL,LOM,IM, AND LON RESPECTIVELY.
34 #
35 # THE SIMPLIFIED POSITION VECTOR OF THE MOON IS
36 # -
37 # M=(COS(LOM), COS(OBL)*SIN(LOM)-SIN(OBL)*SIN(IM)*SIN(LOM-LON), SIN(OBL)*SIN(LOM)+COS(OBL)*SIN(IM)*SIN(LOM-LON))
38 #
39 # WHERE
40 # LOM=LOM +LOM *T-(A *SIN(2PI*T/27.5545)+A *COS(2PI*T/27.5545)+B *SIN(2PI*T/32)+B *COS(2PI*T/32)), AND
41 # 0 R 0 1 0 1
42 # LON=LON +LON
43 # 0 R
44 # A , A , B AND B ARE STORED AS AMOD AND BMOD (SEE DESCRIPTION OF CMOD, ABOVE). COS(OBL), SIN(OBL)*SIN(IM),
45 # 0 1 0 1
46 # SIN(OBL), AND COS(OBL)*SIN(IM) ARE STORED IN KONMAT AS K1, K2, K3 AND K4, RESPECTIVELY. LOM , LOM , LON , LON
47 # ARE STORED AS LOMO, LOMR, LONO, AND LONR IN RATESP. 0 R 0 R
48 # THE THREE PHIS ARE STORED AS AARG, BARG, AND CARG(SUN). ALL CONSTANTS ARE UPDATED BY YEAR.
49 #
50 # CALLING SEQUENCE
```

```
1 # CALL LSPOS. RETURN IS VIA QPRET.
2 # ALARMS OR ABORTS
3 # NONE
4 # ERASABLE INITIALIZATION REQUIRED
5 # TEPHEM - TIME FROM MIGNIGHT 1 JULY PRECEDING THE LAUNCH TO THE TIME OF THE LAUNCH (WHEN THE AGC CLOCK WENT
6 # TO ZERO). TEPHEM IS TP WITH UNITS OF CENTI-SECONDS.
7 # TIME2 AND TIME1 ARE IN MPAC AND MPAC +1 WHEN PROGRAM IS CALLED.
8 # OUTPUT
9 # UNIT POSITIONAL VECTOR OF SUN IN VSUN. (SCALED B-1)
10 # UNIT POSITIONAL VECTOR OF MOON IN VMOON. (SCALED B-1)
11 # SUBROUTINES USED
12 # NONE
13 # DEBRIS
14 # CURRENT CORE SET,WORK AREA AND FREEFLAG
15 BANK 04
16 SETLOC EPHEM
17 BANK
18 EBANK= VSUN
19 COUNT* $$/EPHEM
20 LUNPOS EQUALS LSPOS
21 SETPD SR
22 0
23 14D # TP
24 TAD DDV
25 TEPHEM # TIME OF LAUNCH [IN CENTISEC B 42]
26 CSTODAY # 24 HOURS-8640000 CENTI-SECS/DAY B-33
27 STORE TIMEP # T IN DAYS [@ B 9 = 512 DAYS]
28 AXT,1 AXT,2 # [GRANULARITY ~ 0.164 SEC]
29 0
30 0
31 CLEAR
32 FREEFLAG # SWITCH BIT
33 POSITA DLOAD
34 KONMAT +2 # ZERO$
35 GTMP
36 POSITB DLOAD
37 DMP*
38 TIMEP # T
39 VAL67 +4,1 # 1/27 OR 1/32 OR 1/365
```



|    |        |        |             |                                    |
|----|--------|--------|-------------|------------------------------------|
| 1  |        |        |             |                                    |
| 2  |        | SL     | DAD*        |                                    |
| 3  |        |        | 8D          |                                    |
| 4  |        |        | VAL67 +2,1  | # AARG                             |
| 5  |        | SIN    | DMP*        | # SIN(T/27+PHI) OR T/32 OR T/365   |
| 6  |        |        | VAL67,1     | # (A0**2+A1**2)**1/2SIN(X+PHIA)    |
| 7  |        | DAD    | INCR,1      | # PLUS                             |
| 8  |        |        | GTMP        | # (B0**2+B1**2)**1/2SIN(X+PHIB)    |
| 9  |        | DEC    | -6          |                                    |
| 10 |        | STORE  | GTMP        | # OR (C0**2+C1**2)**1/2SIN(X+PHIC) |
| 11 |        | BOFSET |             |                                    |
| 12 |        |        | FREEFLAG    |                                    |
| 13 |        |        | POSITB      |                                    |
| 14 | POSITD | DLOAD  | DMP*        |                                    |
| 15 |        |        | TIMEP       | # T                                |
| 16 |        |        | RATESP,2    | # LOMR,LOSR,LONR                   |
| 17 |        | SL     | DAD*        |                                    |
| 18 |        |        | 5D          |                                    |
| 19 |        |        | RATESP +6,2 | # LOMO,LOSO,LONO                   |
| 20 |        | DSU    |             |                                    |
| 21 |        |        | GTMP        |                                    |
| 22 |        | STORE  | STMP,2      | # LOM,LOS,LON                      |
| 23 |        | SLOAD  | INCR,2      |                                    |
| 24 |        |        | X2          |                                    |
| 25 |        | DEC    | -2          |                                    |
| 26 |        | DAD    | BZE         |                                    |
| 27 |        |        | RCB-13      | # PLUS 2                           |
| 28 |        |        | POSITE      | # 2ND                              |
| 29 |        | BPL    |             |                                    |
| 30 |        |        | POSITA      | # 1ST                              |
| 31 | POSITF | DLOAD  | DSU         | # 3RD                              |
| 32 |        |        | STMP        | # LOM                              |
| 33 |        |        | STMP +4     | # LON                              |
| 34 |        | SIN    | PDDL        | # SIN(LOM-LON)                     |
| 35 |        |        | STMP        |                                    |
| 36 |        | SIN    | PDDL        | # SIN LOM                          |
| 37 |        |        | STMP        |                                    |
| 38 |        | COS    | VDEF        | # COS LOM                          |
| 39 |        | MXV    | UNIT        |                                    |
| 40 |        |        | KONMAT      | # K1,K2,K3,K4,                     |
| 41 |        | STORE  | VMOON       |                                    |
| 42 |        | DLOAD  | PDDL        |                                    |
| 43 |        |        | KONMAT +2   | # ZERO                             |
| 44 |        |        | STMP +2     |                                    |
| 45 |        | SIN    | PDDL        | # SIN LOS                          |
| 46 |        |        | STMP +2     |                                    |
| 47 |        | COS    | VDEF        | # COS LOS                          |
| 48 |        | MXV    | UNIT        |                                    |
| 49 |        |        | KONMAT      |                                    |
| 50 |        | STORE  | VSUN        |                                    |
| 51 |        | RVQ    |             |                                    |
| 52 |        |        |             |                                    |
| 53 |        |        |             |                                    |
| 54 |        |        |             |                                    |
| 55 |        |        |             |                                    |
| 56 |        |        |             |                                    |
| 57 |        |        |             |                                    |
| 58 |        |        |             |                                    |
| 59 |        |        |             |                                    |
| 60 |        |        |             |                                    |



|    |                               |        |            |                     |    |
|----|-------------------------------|--------|------------|---------------------|----|
| 1  |                               |        |            |                     | 1  |
| 2  | POSITE                        | DLOAD  |            |                     | 2  |
| 3  |                               |        | KONMAT +2  | # ZEROS             | 3  |
| 4  |                               | STORE  | GTMP       |                     | 4  |
| 5  |                               | GOTO   |            |                     | 5  |
| 6  |                               |        | POSITD     |                     | 6  |
| 7  | LUNVEL                        | RVQ    | #          | TO FOOL INTEGRATION | 7  |
| 8  |                               | SETLOC | EPHEM1     |                     | 8  |
| 9  |                               | BANK   |            |                     | 9  |
| 10 |                               |        |            |                     | 10 |
| 11 |                               | COUNT* | \$\$/EPHEM |                     | 11 |
| 12 | STMP                          | EQUALS | 16D        |                     | 12 |
| 13 | GTMP                          | EQUALS | 22D        |                     | 13 |
| 14 | TIMEP                         | EQUALS | 24D        |                     | 14 |
| 15 |                               |        |            |                     | 15 |
| 16 | # *** END OF LEMP50S .115 *** |        |            |                     | 16 |
| 17 |                               |        |            |                     | 17 |
| 18 |                               |        |            |                     | 18 |
| 19 |                               |        |            |                     | 19 |
| 20 |                               |        |            |                     | 20 |
| 21 |                               |        |            |                     | 21 |
| 22 |                               |        |            |                     | 22 |
| 23 |                               |        |            |                     | 23 |
| 24 |                               |        |            |                     | 24 |
| 25 |                               |        |            |                     | 25 |
| 26 |                               |        |            |                     | 26 |
| 27 |                               |        |            |                     | 27 |
| 28 |                               |        |            |                     | 28 |
| 29 |                               |        |            |                     | 29 |
| 30 |                               |        |            |                     | 30 |
| 31 |                               |        |            |                     | 31 |
| 32 |                               |        |            |                     | 32 |
| 33 |                               |        |            |                     | 33 |
| 34 |                               |        |            |                     | 34 |
| 35 |                               |        |            |                     | 35 |
| 36 |                               |        |            |                     | 36 |
| 37 |                               |        |            |                     | 37 |
| 38 |                               |        |            |                     | 38 |
| 39 |                               |        |            |                     | 39 |
| 40 |                               |        |            |                     | 40 |
| 41 |                               |        |            |                     | 41 |
| 42 |                               |        |            |                     | 42 |
| 43 |                               |        |            |                     | 43 |
| 44 |                               |        |            |                     | 44 |
| 45 |                               |        |            |                     | 45 |
| 46 |                               |        |            |                     | 46 |
| 47 |                               |        |            |                     | 47 |
| 48 |                               |        |            |                     | 48 |
| 49 |                               |        |            |                     | 49 |
| 50 |                               |        |            |                     | 50 |
| 51 |                               |        |            |                     | 51 |
| 52 |                               |        |            |                     | 52 |
| 53 |                               |        |            |                     | 53 |
| 54 |                               |        |            |                     | 54 |
| 55 |                               |        |            |                     | 55 |
| 56 |                               |        |            |                     | 56 |
| 57 |                               |        |            |                     | 57 |
| 58 |                               |        |            |                     | 58 |
| 59 |                               |        |            |                     | 59 |
| 60 |                               |        |            |                     | 60 |

```
1 # PROGRAM NAME -- DOWN TELEMETRY PROGRAM
2 # MOD NO. -- 0 TO COMPLETELY REWRITE THE DOWN TELEMETRY PROGRAM AND DOWNLINK ERASABLE DUMP PROGRAM FOR THE
3 # PURPOSE OF SAVING APPROXIMATELY 150 WORDS OF CORE STORAGE.
4 # THIS CHANGE REQUIRES AN ENTIRELY NEW METHOD OF SPECIFYING DOWNLINK LISTS. REFER TO DOWNLINK
5 # LISTS LOG SECTION FOR MORE DETAILS. HOWEVER THIS CHANGE WILL NOT AFFECT THE GROUND PROCESSING
6 # OF DOWN TELEMETRY DATA.
7 # MOD BY -- KILROY, SMITH, DEWITT
8 # DATE -- 02 OCT 67
9 # AUTHORS -- KILROY, SMITH, DWWITT, DEWOLF, FAGIN
10 # LOG SECTION -- DOWN-TELEMETRY PROGRAM
11 #
12 # FUNCTIONAL DESCRIPTION -- THIS ROUTINE IS INITIATED BY TELEMETRY END
13 # PULSE FROM THE DOWNLINK TELEMETRY CONVERTER. THIS PULSE OCCURS
14 # AT 50 TIMES PER SEC (EVERY 20 MS) THEREFORE DODOWNTM IS
15 # EXECUTED AT THESE RATES. THIS ROUTINE SELECTS THE APPROPRIATE
16 # AGC DATA TO BE TRANSMITTED DOWNLINK AND LOADS IT INTO OUTPUT
17 # CHANNELS 34 AND 35. THE INFORMATION IS THEN GATED OUT FROM THE
18 # LGC IN SERIAL FASHION.
19 #
20 # THIS PROGRAM IS CODED FOR A 2 SECOND DOWNLIST. SINCE DOWNRUPTS
21 # OCCUR EVERY 20 MS AND 2 AGC COMPUTER WORDS CAN BE PLACED IN
22 # CHANNELS 34 AND 35 DURING EACH DOWNRUPT THE PROGRAM IS CAPABLE
23 # OF SENDING 200 AGC WORDS EVERY 2 SECONDS.
24 #
25 # CALLING SEQUENCE -- NONE
26 # PROGRAM IS ENTERED VIA TCF DODOWNTM WHICH IS EXECUTED AS A
27 # RESULT OF A DOWNRUPT. CONTROL IS RETURNED VIA TCF RESUME WHICH
28 # IN EFFECT IS A RESUME.
29 #
30 # SUBROUTINES CALLED -- NONE
31 #
32 # NORMAL EXIT MODE -- TCF RESUME
33 #
34 # ALARM OR ABORT EXIT MODE -- NONE
35 #
36 # RESTART PROTECTION:
37 # ON A FRESH START AND RESTART THE 'STARTSUB' SUBROUTINE WILL INITIALIZE THE DOWNLIST POINTER (ACTUALLY
38 # DNTMGOTO) TO THE BEGINNING OF THE CURRENT DOWNLIST (I.E., CURRENT CONTENTS OF DNLSTADR). THIS HAS THE
39 # EFFECT OF IGNORING THE REMAINDER OF THE DOWNLIST WHICH THE DOWN-TELEMETRY PROGRAM WAS WORKING ON WHEN
40 # THE RESTART (OR FRESH START) OCCURRED AND RESUME DOWN TELEMETRY FROM THE BEGINNING OF THE CURRENT
41 # DOWNLIST.
42 #
43 # ALSO OF INTEREST IS THE FACT THAT ON A RESTART THE AGC WILL ZERO DOWNLINK CHANNELS 13, 34 AND 35.
44 #
45 # DOWNLINK LIST SELECTION:
46 # THE APPROPRIATE DOWNLINK LISTS ARE SELECTED BY THE FOLLOWING:
47 # 1. FRESH START
48 # 2. V37EXXE WHERE XX = THE MAJOR MODE BEING SELECTED.
49 # 3. UPDATE PROGRAM (P27)
50 # 4. NON-V37 SELECTABLE TYPE PROGRAMS (E.G., AGS INITIALIZATION (SUNDANCE, LUMINARY) AND P61-P62
51 # TRANSITION (COLOSSUS) ETC.).
52 #
53 # DOWNLINK LIST RULES AND LIMITATIONS:
54 # READ SECTION(S) WHICH FOLLOW 'DEBRIS' WRITEUP.
55 #
56 # OUTPUT -- EVERY 2 SECONDS 100 DOUBLE PRECISION WORDS (I.E., 200 LGC
57 # COMPUTER WORDS) ARE TRANSMITTED VIA DOWNLINK.
58 #
59 # ERASABLE INITIALIZATION REQUIRED -- NONE
```

```
`DNTMGOTO' AND `DNLSTADR' ARE INITIALIZED BY THE FRESH START PROGRAM.
```

```
#
DEBRIS (ERASABLE LOCATIONS DESTROYED BY THIS PROGRAM) --
LDATALST, DNTMBUFF TO DNTMBUFF +21D, TMINDEX, DNQ.
```

1412THE

1



```
1 # DODOWNTM IS ENTERED EVERY 20 MS BY AN INTERRUPT TRIGGERED BY THE
2 # RECEIPT OF AN ENDPULSE FROM THE SPACECRAFT TELEMETRY PROGRAMMER.
3
4 #
5 # NOTES REGARDING DOWNLINK LISTS ASSOCIATED WITH THIS PROGRAM:
6 # 1. DOWNLISTS. DOWNLISTS MUST BE COMPILED IN THE SAME BANK AS THE
7 # DOWN-TELEMETRY PROGRAM. THIS IS DONE FOR EASE OF CODING, FASTER
8 # EXECUTION.
9 # 2. EACH DOWNLINK LIST CONSISTS OF A CONTROL LIST AND A NUMBER OF
10 # SUBLISTS.
11 # 3. A SUBLIST REFERS TO A SNAPSHOT OR DATA COMMON TO THE SAME OR OTHER
12 # DOWNLINK LISTS. ANY SUBLIST CONTAINING COMMON DATA NEEDS TO BE
13 # CODED ONLY ONCE FOR THE APPLICABLE DOWNLINK LISTS.
14 # 4. SNAPSHOT SUBLISTS REFER SPECIFICALLY TO HOMOGENEOUS DATA WHICH MUST BE
15 # SAVED IN A BUFFER DURING ONE DOWNRUPT.
16 # 5. THE 1DNADR FOR THE 1ST WORD OF SNAPSHOT DATA IS FOUND AT THE END
17 # OF EACH SNAPSHOT SUBLIST, SINCE THE PROGRAM CODING SENDS THIS DP WORD
18 # IMMEDIATELY AFTER STORING THE OTHERS IN THE SNAPSHOT BUFFER.
19 # 6. ALL LISTS ARE COMBINATIONS OF CODED ERASABLE ADDRESS CONSTANTS
20 # CREATED FOR THE DOWNLIST PROGRAM.
21 # A. 1DNADR 1-WORD DOWNLIST ADDRESS.
22 # SAME AS ECADR, BUT USED WHEN THE WORD ADDRESSED IS THE LEFT
23 # HALF OF A DOUBLE-PRECISION WORD FOR DOWN TELEMETRY.
24 # B. 2DNADR - 6DNADR N-WORD DOWNLIST ADDRESS, N = 2 - 6.
25 # SAME AS 1DNADR, BUT WITH THE 4 UNUSED BITS OF THE ECADR FORMAT
26 # FILLED IN WITH 0001-0101. USED TO POINT TO A LIST OF N DOUBLE-
27 # PRECISION WORDS, STORED CONSECUTIVELY, FOR DOWN TELEMETRY.
28 # C. DNCHAN DOWNLIST CHANNEL ADDRESS.
29 # SAME AS 1DNADR, BUT WITH PREFIX BITS 0111. USED TO POINT TO
30 # A PAIR OF CHANNELS FOR DOWN TELEMETRY.
31 # D. DNPTR DOWN-TELEMETRY SUBLIST POINTER.
32 # SAME AS CAF BUT TAGGED AS A CONSTANT. USED IN CONTROL LIST TO POINT TO A SUBLIST.
33 # CAUTION --- A DNPTR CANNOT BE USED IN A SUBLIST.
34 # 7. THE WORD ORDER CODE IS SET TO ZERO AT THE BEGINNING OF EACH DOWNLIST (I.E., CONTROL LIST) AND WHEN
35 # A '1DNADR TIME2' IS DETECTED IN THE CONTROL LIST (ONLY).
36 # 8. IN THE SNAPSHOT SUBLIST ONLY, THE DNADR'S CANNOT POINT TO THE FIRST WORD OF ANY EBANK.
37
38 # DOWNLIST LIST RESTRICTIONS:
39 # (THE FOLLOWING POINTS MAY BE LISTED ELSEWHERE BUT ARE LISTED HERE SO IT IS CLEAR THAT THESE THINGS CANNOT BE
40 # DONE)
41 # 1. SNAPSHOT DOWNLIST:
42 # (A) CANNOT CONTAIN THE FOLLOWING ECADRS (I.E., 1DNADR'S): Q, 400, 1000, 1400, 2000, 2400, 3000, 3400.
43 # (B) CAN CONTAIN ONLY 1DNADR'S
44 # 2. ALL DOWNLINKED DATA (EXCEPT CHANNELS) IS PICKED UP BY A DCA SO DOWNLINK LISTS CANNOT CONTAIN THE
45 # EQUIVALENT OF THE FOLLOWING ECADRS (I.E., 1DNADRS): 377, 777, 1377, 1777, 2377, 2777, 3377, 3777.
46 # (NOTE: THE TERM 'EQUIVALENT' MEANT THAT THE 1DNADR TO 6DNADR WILL BE PROCESSED LIKE 1 TO 6 ECADRS)
47 # 3. CONTROL LISTS AND SUBLISTS CANNOT HAVE ENTRIES = OCTAL 00000 OR OCTAL 77777
```

# 4. THE '1DNADR TIME2' WHICH WILL CAUSE THE DOWNLINK PROGRAM TO SET THE WORDER CODE TO 3 MUST APPEAR IN THE  
# CONTROL SECTION OF THE DOWNLIST.

# 5. 'DNCHAN 0' CANNOT BE USED.

# 6. 'DNPTR 0' CANNOT BE USED.

# 7. DNPTR CANNOT APPEAR IN A SUBLIST.

# EBANK SETTINGS

# IN THE PROCESS OF SETTING THE EBANK (WHEN PICKING UP DOWNLINK DATA) THE DOWN TELEMETRY PROGRAM PUTS

# 'GARBAGE' INTO BITS15-12 OF EBANK. HUGH BLAIR-SMITH WARNS US THAT BITS15-12 OF EBANK MAY BECOME

# SIGNIFICANT SOMEDAY IN THE FUTURE. IF/WHEN THAT HAPPENS, THE PROGRAM SHOULD INSURE (BY MASKING ETC.)

# THAT BITS 15-12 OF EBANK ARE ZERO.

# INITIALIZATION REQUIRED -- TO INTERRUPT CURRENT LIST AND START A NEW ONE.

# 1. ADRES OF DOWNLINK LIST INTO DNLSTADR

# 2. NEGONE INTO SUBLIST

# 3. NEGONE INTO DNECADR

BANK 22  
SETLOC DOWNTLM  
BANK

EBANK= DNTMBUFF

DODOWNTM COUNT\* \$\$/DPROG  
TS BANKRUPT  
EXTEND

QXCH QRUPT # SAVE Q

TCF WOTEST

W01 EXTEND # SET WORD ORDER BIT TO 1 ONLY IF IT

WOR CHAN13 # ALREADY ISN'T

TC DNTMGOTO # GOTO APPROPRIATE PHASE OF PROGRAM

DNPHASE1 CA NEGONE # INITIALIZE ALL CONTROL WORDS

TS SUBLIST # WORDS TO MINUS ONE

TS DNECADR

CA LDNPHAS2 # SET DNTMGOTO = 0 ALL SUSEQUENT DOWRUPTS

TS DNTMGOTO # GO TO DNPHASE2

TCF NEWLIST

DNPHASE2 CCS DNECADR # SENDING OF DATA IN PROGRESS

DODNADR TC FETCH2WD # YES -- THEN FETCH THE NEXT 2 SP WORDS

MINTIME2 -1DNADR TIME2 # NEGATIVE OF TIME2 1DNADR

TCF +1 # (ECADR OF 3776 + 74001 = 77777)

CCS SUBLIST # IS THE SUBLIST IN CONTROL

TCF NEXTINSL # YES

|    |          |        |           |                                            |
|----|----------|--------|-----------|--------------------------------------------|
| 1  | DNADRDCR | OCT    | 74001     | # DNADR COUNT AND ECADR DECREMENTER        |
| 2  |          |        |           |                                            |
| 3  | CHKLIST  | CA     | CTLIST    |                                            |
| 4  |          | EXTEND |           |                                            |
| 5  |          | BZMF   | NEWLIST   | # IT WILL BE NEGATIVE AT END OF LIST       |
| 6  |          |        |           |                                            |
| 7  | NEWLIST  | TCF    | NEXTINCL  |                                            |
| 8  |          | INDEX  | DNLSTCOD  |                                            |
| 9  |          | CA     | DNTABLE   | # INITIALIZE CTLIST WITH                   |
| 10 |          | TS     | CTLIST    | # STARTING ADDRESS OF NEW LIST             |
| 11 |          | CS     | DNLSTCOD  |                                            |
| 12 |          | TCF    | SENDID +3 |                                            |
| 13 | NEXTINCL | INDEX  | CTLIST    |                                            |
| 14 |          | CA     | 0         |                                            |
| 15 |          | CCS    | A         |                                            |
| 16 |          | INCR   | CTLIST    | # SET POINTER TO PICK UP NEXT CTLIST WORD  |
| 17 |          | TCF    | +4        | # ON NEXT ENTRY TO PROG. (A SHOULD NOT =0) |
| 18 |          | XCH    | CTLIST    | # SET CTLIST TO NEGATIVE AND PLACE(CODING) |
| 19 |          | COM    |           | # UNCOMPLEMENTED DNADR INTO A. (FOR LA)    |
| 20 |          | XCH    | CTLIST    | # (ST IN )                                 |
| 21 | +4       | INCR   | A         | # (CTLIST)                                 |
| 22 |          | TS     | DNECADR   | # SAVE DNADR                               |
| 23 |          | AD     | MINTIME2  | # TEST FOR TIME2 (NEG. OF ECADR)           |
| 24 |          | CCS    | A         |                                            |
| 25 |          | TCF    | SETWO +1  | # DON'T SET WORD ORDER CODE                |
| 26 | MINB1314 | OCT    | 47777     | # MINUS BIT 13 AND 14 (CAN'T GET HERE)     |
| 27 |          | TCF    | SETWO +1  | # DON'T SET WORD ORDER CODE                |
| 28 | SETWO    | TC     | WOZERO    | # GO SET WORD ORDER CODE TO ZERO.          |
| 29 | +1       | CA     | DNECADR   | # RELOAD A WITH THE DNADR.                 |
| 30 | +2       | AD     | MINB1314  | # IS THIS A REGULAR DNADR?                 |
| 31 |          | EXTEND |           |                                            |
| 32 |          | BZMF   | FETCH2WD  | # YES. (A MUST NEVER BE ZERO)              |
| 33 |          | AD     | MINB12    | # NO. IS IT A POINTER (DNPTR) OR A         |
| 34 |          | EXTEND |           | # CHANNEL(DNCHAN)                          |
| 35 |          | BZMF   | DODNPTR   | # IT'S A POINTER. (A MUST NEVER BE ZERO)   |
| 36 |          |        |           |                                            |
| 37 | DODNCHAN | TC     | 6         | # (EXECUTED AS EXTEND) IT'S A CHANNEL      |
| 38 |          | INDEX  | DNECADR   |                                            |
| 39 |          | INDEX  | 0 -4000   | # (EXECUTED AS READ)                       |
| 40 |          | TS     | L         |                                            |
| 41 |          | TC     | 6         | # (EXECUTED AS EXTEND)                     |
| 42 |          | INDEX  | DNECADR   |                                            |
| 43 |          | INDEX  | 0 -4001   | # (EXECUTED AS READ)                       |
| 44 |          | TS     | DNECADR   | # SET DNECADR                              |
| 45 |          | CA     | NEGONE    | # TO MINUS                                 |
| 46 |          | XCH    | DNECADR   | # WHILE PRESERVING A.                      |
| 47 |          | TCF    | DNTMEXIT  | # GO SEND CHANNELS                         |
| 48 |          |        |           |                                            |
| 49 | WOZERO   | CS     | BIT7      |                                            |
| 50 |          | EXTEND |           |                                            |
| 51 |          | WAND   | CHAN13    | # SET WORD ORDER CODE TO ZERO              |
| 52 |          |        |           |                                            |
| 53 |          |        |           |                                            |
| 54 |          |        |           |                                            |
| 55 |          |        |           |                                            |
| 56 |          |        |           |                                            |
| 57 |          |        |           |                                            |
| 58 |          |        |           |                                            |
| 59 |          |        |           |                                            |
| 60 |          |        |           |                                            |



```
1
2 TC Q # RETURN TO CALLER
3
4 DODNPTR INDEX DNECADR # DNECADR CONTAINS ADRES OF SUBLIST
5 0 0 # CLEAR AND ADD LIST ENTRY INTO A.
6 CCS A # IS THIS A SNAPSHOT SUBLIST
7
8 CA DNECADR # NO, IT IS A REGULAR SUBLIST.
9 TCF DOSUBLST # A MUST NOT BE ZERO.
10
11 XCH DNECADR # YES. IT IS A SNAPSHOT SUBLIST.
12 TS SUBLIST # C(DNECADR) INTO SUBLIST
13 CAF ZERO # A INTO A
14 XCH TMINDEX # (NOTE: TMINDEX = DNECADR)
15
16 # THE FOLLOWING CODING (FROM SNAPLOOP TO SNAPEND) IS FOR THE PURPOSE OF TAKING A SNAPSHOT OF 12 DP REGISTERS.
17 # THIS IS DONE BY SAVING 11 DP REGISTERS IN DNTMBUFF AND SENDING THE FIRST DP WORD IMMEDIATELY.
18 # THE SNAPSHOT PROCESSING IS THE MOST TIME CONSUMING AND THEREFORE THE CODING AND LIST STRUCTURE WERE DESIGNED
19 # TO MINIMIZE TIME. THE TIME OPTIMIZATION RESULTS IN RULES UNIQUE TO THE SNAPSHOT PORTION OF THE DOWNLIST.
20 # THESE RULES ARE
21 # 1. ONLY 1DNADR'S CAN APPEAR IN THE SNAPSHOT SUBLIST
22 # 2. THE 1DNADR'S CANNOT REFER TO THE FIRST LOCATION IN ANY BANK.
23
24 SNAPLOOP TS EBANK # SET EBANK
25 MASK LOW8 # ISOLATE RELATIVE ADDRESS
26
27 EXTEND
28 INDEX A
29 EBANK= 1401
30
31 DCA 1401 # PICK UP 2 SNAPSHOT WORDS.
32 EBANK= DNTMBUFF
33 INDEX TMINDEX
34
35 DXCH DNTMBUFF # STORE 2 SNAPSHOT WORDS IN BUFFER
36 INCR TMINDEX # SET BUFFER INDEX FOR NEXT 2 WORDS.
37 INCR TMINDEX
38
39 SNAPAGN INCR SUBLIST # SET POINTER TO NEXT 2 WORDS OF SNAPSHOT
40 INDEX SUBLIST
41 0 0 # = CA SSSS (SSSS = NEXT ENTRY IN SUBLIST)
42
43 CCS A # TEST FOR LAST TWO WORDS OF SNAPSHOT.
44 TCF SNAPLOOP # NOT LAST TWO.
45
46 LDNPHAS2 GENADR DNPHASE2
47
48 TS SUBLIST # YES, LAST. SAVE A.
49 CA NEGONE # SET DNECADR AND
50 TS DNECADR # SUBLIST POINTERS
51
52 XCH SUBLIST #
53 TS EBANK #
54 MASK LOW8 #
55
56 EXTEND
57 INDEX A
58 EBANK= 1401
59
60 DCA 1401 # PICK UP FIRST 2 WORDS OF SNAPSHOT.
```

```
1
2 SNAPEND EBANK= DNTMBUFF
3 TCF DNTMEXIT # NOW TO SEND THEM.
4
5 FETCH2WD CA DNECADR
6 TS EBANK # SET EBANK
7
8 MASK LOW8 # ISOLATE RELATIVE ADDRESS
9 TS L
10 CA DNADRDCR # DECREMENT COUNT AND ECADR
11
12 ADS DNECADR
13 EXTEND
14 INDEX L
15
16 EBANK= 1400
17 DCA 1400 # PICK UP 2 DATA WORDS
18 EBANK= DNTMBUFF
19 TCF DNTMEXIT # NOW GO SEND THEM.
20
21 DOSUBLST TS SUBLIST # SET SUBLIST POINTER
22 NEXTINSL INDEX SUBLIST
23 0 0 # = CA SSSS (SSSS = NEXT ENTRY IN SUBLIST)
24 CCS A # IS IT THE END OF THE SUBLIST
25
26 INCR SUBLIST # NO --
27 TCF +4
28 TS SUBLIST # SAVE A.
29
30 CA NEGONE # SET SUBLIST TO MINUS
31 XCH SUBLIST # RETRIEVE A.
32 +4 A
33
34 TS DNECADR # SAVE DNADR
35 TCF SETWO +2 # GO USE COMMON CODING (PROLEMS WOULD
36 # OCCUR IF THE PROGRAM ENCOUNTERED A
37 # DNPTR NOW)
38
39 DNTMEXIT EXTEND # DOWN-TELEMETRY EXIT
40
41 WRITE DNTM1 # TO SEND A + L TO CHANNELS 34 + 35
42 CA L # RESPECTIVELY
43
44 TMEXITL EXTEND
45
46 WRITE DNTM2
47 TCF RESUME # EXIT TELEMETRY PROGRAM VIA RESUME.
48
49 MINB12 EQUALS -1/8
50 DNECADR EQUALS TMINDEX
51 CTLIST EQUALS LDATALST
52 SUBLIST EQUALS DNQ
```

# SUBROUTINE NAME -- DNDUMP

# FUNCTIONAL DESCRIPTION -- TO SEND (DUMP) ALL ERASABLE STORAGE 'N' TIMES. (N=1 TO 4). BANKS ARE SENT ONE AT A TIME  
# EACH BANK IS PRECEDED BY AN ID WORD, SYNCH BITS, ECADR AND TIME1 FOLLOWED BY THE 256D WORDS OF EACH  
# EBANK. EBANKS ARE DUMPED IN ORDER (I.E., EBANK 0 FIRST, THEN EBANK1 ETC.)

# CALLING SEQUENCE -- THE GROUND OR ASTRONAUT BY KEYING V74E CAN INITIALIZE THE DUMP.  
# AFTER KEYING IN V74E THE CURRENT DOWNLIST WILL BE IMMEDIATELY TERMINATED AND THE DOWNLINK ERASABLE DUMP  
# WILL BEGIN.

# ONCE INITIATED THE DOWNLINK ERASABLE DUMP CAN BE TERMINATED (AND INTERRUPTED DOWNLIST REINSTATED) ONLY  
# BY THE FOLLOWING:

- # 1. A FRESH START
- # 2. COMPLETION OF ALL DOWNLINK DUMPS REQUESTED (ACCORDING TO BITS SET IN DUMPCNT). NOTE THAT DUMPCNT  
# CAN BE ALTERED BY A V21N01.
- # 3. AND INVOLUNTARILY BY A RESTART.

# NORMAL EXIT MODE -- TCF DNPHASE1

# ALARM OR ABORT MODE -- NONE

# \*SUBROUTINES CALLED -- NONE

# ERASABLE INITIALIZATION REQUIRED --

|   |         |           |                                          |
|---|---------|-----------|------------------------------------------|
| # | DUMPCNT | OCT 20000 | IF 4 COMPLETE ERASABLE DUMPS ARE DESIRED |
| # | DUMPCNT | OCT 10000 | IF 2 COMPLETE ERASABLE DUMPS ARE DESIRED |
| # | DUMPCNT | OCT 04000 | IF 1 COMPLETE ERASABLE DUMP IS DESIRED   |

# DEBRIS -- DUMPLOC, DUMPSW, DNTMGOTO, EBANK, AND CENTRAL REGISTERS

# TIMING --  $TIME (IN SECS) = ((NO.DUMPS)*(NO.EBANKS)*(WDSPEREBANK + NO.IDWDS)) / NO.WDSPERSEC$   
#  $TIME (IN SECS) = ( 4 ) * ( 8 ) * ( 256 + 4 ) / 100$   
# THUS TIME (IN SECS TO SEND DUMP OF ERASABLE 4 TIMES VIA DOWNLINK) = 83.2 SECONDS

# STRUCTURE OF ONE EBANK AS IT IS SENT BY DOWNLINK PROGRAM --  
# (REMINDER -- THIS ONLY DESCRIBES ONE OF THE 8 EBANKS X 4 (DUMPS) = 32 EBANKS WHICH WILL BE SENT BY DNDUMP)

| # | DOWNLIST                    | W       |                                                                |
|---|-----------------------------|---------|----------------------------------------------------------------|
| # | WORD TAKEN FROM CONTENTS OF | EXAMPLE | COMMENTS                                                       |
| # | 1 ERASID                    | 0177X 0 | DOWNLIST I.D. FOR DOWNLINK ERASABLE DUMP (X=7 CSM, 6 LM)       |
| # | 2 LOWIDCOD                  | 77340 1 | DOWNLINK SYNCH BITS. (SAME ONE USED IN ALL OTHER DOWNLISTS)    |
| # | 3 DUMPLOC                   | 13400 1 | (SEE NOTES ON DUMPLOC) 1 = 3RD ERAS DUMP, 3400=ECADR OF 5TH WD |
| # | 4 TIME1                     | 14120 1 | TIME IN CENTISECONDS                                           |
| # | 5 FIRST WORD OF EBANK X     | 03400 1 | IN THIS EXAMPLE THIS WORD = CONTENTS OF E7,1400 (ECADR 3400)   |
| # | 6 2ND WORD OF EBANK X       | 00142 1 | IN THIS EXAMPLE THIS WORD = CONTENTS OF E7,1401 (ECADR 3401)   |
| # | 7 3RD WORD OF EBANK X       | 00142 1 | IN THIS EXAMPLE THIS WORD = CONTENTS OF E7,1402 (ECADR 3402)   |

|   |      |                       |                                                                      |
|---|------|-----------------------|----------------------------------------------------------------------|
| # | .    |                       |                                                                      |
| # | .    |                       |                                                                      |
| # | .    |                       |                                                                      |
| # | 260D | 256TH WORD OF EBANK X | 03777 1 IN THIS EXAMPLE THIS WORD = CONTENTS OF E7,1777 (ECADR 3777) |

# NOTE -- DUMPLOC CONTAINS THE COUNTER AND ECADR FOR EACH WORD BEING SENT.

# THE BIT STRUCTURE OF DUMPLOC IS FOLLOW --

|   |                    |                                                                                                                                                |
|---|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| # |                    | X = NOT USED                                                                                                                                   |
| # | X ABC EEE RRRRRRRR | ABC = ERASABLE DUMP COUNTER (I.E. ABC = 0,1,2, OR 3 WHICH MEANS THAT<br>COMPLETE ERASABLE DUMP NUMBER 1,2,3, OR 4 RESPECTIVELY IS IN PROGRESS) |
| # |                    | EEE = EBANK BITS                                                                                                                               |
| # |                    | RRRRRRRR = RELATIVE ADDRESS WITHIN AN EBANK                                                                                                    |



```
1 DNDUMPI CA ZERO # INITIALIZE DOWNLINK
2 TS DUMPLOC # ERASABLE DUMP
3
4 +2 TC SENDID # GO SEND ID AND SYNCH BITS
5 CA LDNDUMP1 # SET DNTMGOTO
6 TS DNTMGOTO # TO LOCATION FOR NEXT PASS
7
8 CA TIME1 # PLACE TIME1
9 XCH L # INTO L
10 CA DUMPLOC # AND ECADR OF THIS EBANK INTO A
11 TCF DNTMEXIT # SEND DUMPLOC AND TIME1
12
13 LDNDUMP ADRES DNDUMP
14 LDNDUMP1 ADRES DNDUMP1
15
16 DNDUMP CA TWO # INCREMENT ECADR IN DUMPLOC
17 ADS DUMPLOC # TO NEXT DP WORD TO BE
18 MASK LOW8 # DUMPED AND SAVE IT.
19 CCS A # IS THIS THE BEGINNING OF A NEW EBANK
20 TCF DNDUMP2 # NO -- THEN CONTINUE DUMPING
21 CA DUMPLOC # YES -- IS THIS THE END OF THE
22 MASK DUMPCNT # N TH (N = 1 TO 4) COMPLETE ERASABLE
23 MASK PRI034 # DUMP (BIT14 FOR 4, BIT13 FOR 2 OR BIT12
24 CCS A # FOR 1 COMPLETE ERASABLE DUMP(S)).
25 TCF DNPHASE1 # YES -- START SENDING INTERRUPTED DOWNLIST
26
27 TCF DNDUMPI +2 # AGAIN
28
29 DNDUMP1 CA LDNDUMP # SET DNTMGOTO
30 TS DNTMGOTO # FOR WORDS 3 TO 256D OF CURRENT EBANK
31
32 DNDUMP2 CA DUMPLOC # SET EBANK
33 TS EBANK # ISOLATE RELATIVE ADDRESS.
34 MASK LOW8
35 TS Q # (NOTE: MASK INSTRUCTION IS USED TO PICK
36 CA NEG0 # UP ERASABLE REGISTERS SO THAT EDITING
37 TS L # REGISTERS 20-23 WILL NOT BE ALTERED.)
38 INDEX Q
39 EBANK= 1400 # PICK UP LOW ORDER REGISTER OF PAIR
40 MASK 1401 # OF ERASABLE REGISTERS.
41 XCH L
42 INDEX Q # PICK UP HIGH ORDER REGISTER OF PAIR
43 MASK 1400 # OF ERASABLE REGISTERS.
44 EBANK= DNTMBUFF
45 TCF DNTMEXIT # GO SEND THEM
46
47 SENDID EXTEND # ** ENTRANCE USED BY ERASABLE DUMP PROG. **
48 QXCH DNTMGOTO # SET DNTMGOTO SO NEXT TIME PROG WILL GO
49 CAF ERASID # TO LOCATION FOLLOWING `TC SENDID'
50
51 TS L # ** ENTRANCE USED BY REGULAR DOWNLINK PG **
```



|    |        |        |          |                                         |    |
|----|--------|--------|----------|-----------------------------------------|----|
| 1  |        |        |          |                                         | 1  |
| 2  |        | TC     | WOZERO   | # GO SET WORD ORDER CODE TO ZERO        | 2  |
| 3  |        | CAF    | LOWIDCOD | # PLACE SPECIAL ID CODE INTO L          | 3  |
| 4  |        | XCH    | L        | # AND ID BACK INTO A                    | 4  |
| 5  |        | TCF    | DNTMEXIT | # SEND DOWNLIST ID CODE(S).             | 5  |
| 6  |        |        |          |                                         | 6  |
| 7  | WOTEST | CA     | BIT7     | # AT THE BEGINNING OF THE LIST THE WORD | 7  |
| 8  |        | EXTEND |          | # ORDER BIT WILL BE SET BACK TO ZERO    | 8  |
| 9  |        | RAND   | CHAN13   |                                         | 9  |
| 10 |        | CCS    | A        |                                         | 10 |
| 11 |        | TC     | DNTMGOTO |                                         | 11 |
| 12 |        | CA     | BIT7     |                                         | 12 |
| 13 |        | TCF    | WO1      |                                         | 13 |
| 14 |        |        |          |                                         | 14 |
| 15 |        |        |          |                                         | 15 |
| 16 |        |        |          |                                         | 16 |
| 17 |        |        |          |                                         | 17 |
| 18 |        |        |          |                                         | 18 |
| 19 |        |        |          |                                         | 19 |
| 20 |        |        |          |                                         | 20 |
| 21 |        |        |          |                                         | 21 |
| 22 |        |        |          |                                         | 22 |
| 23 |        |        |          |                                         | 23 |
| 24 |        |        |          |                                         | 24 |
| 25 |        |        |          |                                         | 25 |
| 26 |        |        |          |                                         | 26 |
| 27 |        |        |          |                                         | 27 |
| 28 |        |        |          |                                         | 28 |
| 29 |        |        |          |                                         | 29 |
| 30 |        |        |          |                                         | 30 |
| 31 |        |        |          |                                         | 31 |
| 32 |        |        |          |                                         | 32 |
| 33 |        |        |          |                                         | 33 |
| 34 |        |        |          |                                         | 34 |
| 35 |        |        |          |                                         | 35 |
| 36 |        |        |          |                                         | 36 |
| 37 |        |        |          |                                         | 37 |
| 38 |        |        |          |                                         | 38 |
| 39 |        |        |          |                                         | 39 |
| 40 |        |        |          |                                         | 40 |
| 41 |        |        |          |                                         | 41 |
| 42 |        |        |          |                                         | 42 |
| 43 |        |        |          |                                         | 43 |
| 44 |        |        |          |                                         | 44 |
| 45 |        |        |          |                                         | 45 |
| 46 |        |        |          |                                         | 46 |
| 47 |        |        |          |                                         | 47 |
| 48 |        |        |          |                                         | 48 |
| 49 |        |        |          |                                         | 49 |
| 50 |        |        |          |                                         | 50 |
| 51 |        |        |          |                                         | 51 |
| 52 |        |        |          |                                         | 52 |
| 53 |        |        |          |                                         | 53 |
| 54 |        |        |          |                                         | 54 |
| 55 |        |        |          |                                         | 55 |
| 56 |        |        |          |                                         | 56 |
| 57 |        |        |          |                                         | 57 |
| 58 |        |        |          |                                         | 58 |
| 59 |        |        |          |                                         | 59 |
| 60 |        |        |          |                                         | 60 |

# THE FOLLOWING ROUTINE CAN BE USED TO CALL A SUBROUTINE IN ANOTHER BANK. IN THE BANKCALL VERSION, THE  
# CADR OF THE SUBROUTINE IMMEDIATELY FOLLOWS THE TC BANKCALL INSTRUCTION, WITH C(A) AND C(L) PRESERVED.

|          |        |           |  |                                            |
|----------|--------|-----------|--|--------------------------------------------|
|          | BLOCK  | 02        |  |                                            |
|          | COUNT* | \$\$/BANK |  |                                            |
| BANKCALL | DXCH   | BUF2      |  | # SAVE INCOMING A,L.                       |
|          | INDEX  | Q         |  | # PICK UP CADR.                            |
|          | CA     | 0         |  |                                            |
|          | INCR   | Q         |  | # SO WE RETURN TO THE LOC. AFTER THE CADR. |

# SWCALL IS IDENTICAL TO BANKCALL, EXCEPT THAT THE CADR ARRIVES IN A.

|        |       |       |  |                                           |
|--------|-------|-------|--|-------------------------------------------|
| SWCALL | TS    | L     |  |                                           |
|        | LXCH  | FBANK |  | # SWITCH BANKS, SAVING RETURN.            |
|        | MASK  | LOW10 |  | # GET SUB-ADDRESS OF CADR.                |
|        | XCH   | Q     |  | # A,L NOW CONTAINS DP RETURN.             |
|        | DXCH  | BUF2  |  | # RESTORING INPUTS IF THIS IS A BANKCALL. |
|        | INDEX | Q     |  |                                           |
|        | TC    | 10000 |  | # SETTING Q TO SWRETURN.                  |

|          |     |       |    |                                          |
|----------|-----|-------|----|------------------------------------------|
| SWRETURN | XCH | BUF2  | +1 | # COMES HERE TO RETURN TO CALLER. C(A,L) |
|          | XCH | FBANK |    | # ARE PRESERVED FOR RETURN.              |
|          | XCH | BUF2  | +1 |                                          |
|          | TC  | BUF2  |    |                                          |

# THE FOLLOWING ROUTINE CAN BE USED AS A UNILATERAL JUMP WITH C(A,L) PRESERVED AND THE CADR IMMEDIATELY  
# FOLLOWING THE TC POSTJUMP INSTRUCTION.

|          |       |   |  |                       |
|----------|-------|---|--|-----------------------|
| POSTJUMP | XCH   | Q |  | # SAVE INCOMING C(A). |
|          | INDEX | A |  | # GET CADR.           |
|          | CA    | 0 |  |                       |

# BANKJUMP IS THE SAME AS POSTJUMP, EXCEPT THAT THE CADR ARRIVES IN A.

|          |       |       |  |                                      |
|----------|-------|-------|--|--------------------------------------|
| BANKJUMP | TS    | FBANK |  |                                      |
|          | MASK  | LOW10 |  |                                      |
|          | XCH   | Q     |  | # RESTORING INPUT C(A) IF THIS WAS A |
| Q+10000  | INDEX | Q     |  | # POSTJUMP.                          |
| PRI012   | TCF   | 10000 |  | # PRI012 = TCF 10000 = 12000         |

# THE FOLLOWING ROUTINE GETS THE RETURN CADR SAVED BY SWCALL OR BANKCALL AND LEAVES IT IN A.

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| MAKECADR | CAF    | LOW10    |                                            |
|          | MASK   | BUF2     |                                            |
|          | AD     | BUF2     | +1                                         |
|          | TC     | Q        |                                            |
| SUPDACAL | TS     | MPTMP    |                                            |
|          | XCH    | FBANK    | # SET FBANK FOR DATA.                      |
|          | EXTEND |          |                                            |
|          | ROR    | SUPERBNK | # SAVE FBANK IN BITS 15-11, AND            |
|          | XCH    | MPTMP    | # SUPERBANK IN BITS 7-5.                   |
|          | MASK   | LOW10    |                                            |
|          | XCH    | L        | # SAVE REL. ADR. IN BANK, FETCH SUPERBITS. |
|          | INHINT |          | # BECAUSE RUPT DOES NOT SAVE SUPERBANK.    |
|          | EXTEND |          |                                            |
|          | WRITE  | SUPERBNK | # SET SUPERBANK FOR DATA.                  |
|          | INDEX  | L        |                                            |
|          | CA     | 10000    | # PINBALL (FIX MEM DISP) PREVENTS DCA HERE |
|          | XCH    | MPTMP    | # SAVE 1ST WD, FETCH OLD FBANK AND SBANK.  |
|          | EXTEND |          |                                            |
|          | WRITE  | SUPERBNK | # RESTORE SUPERBANK.                       |
|          | RELINT |          |                                            |
|          | TS     | FBANK    | # RESTORE FBANK.                           |
|          | CA     | MPTMP    | # RECOVER FIRST WORD OF DATA.              |
|          | RETURN |          | # 24 WDS. DATACALL 516 MU, SUPDACAL 432 MU |



# THE FOLLOWING ROUTINES ARE IDENTICAL TO BANKCALL AND SWCALL EXCEPT THAT THEY ARE USED IN INTERRUPT.

IBNKCALL DXCH RUPTREG3 # USES RUPTREG3,4 FOR DP RETURN ADDRESS.

INDEX Q  
CAF 0

INCR Q

ISWCALLL TS L

LXCH FBANK  
MASK LOW10  
XCH Q

DXCH RUPTREG3  
INDEX Q  
TC 10000

ISWRETRN XCH RUPTREG4

XCH FBANK

XCH RUPTREG4  
TC RUPTREG3

# 2. USPRCADR ACCESSES INTERPRETIVE CODING IN OTHER THAN THE USER'S FBANK. THE CALLING SEQUENCE IS AS FOLLOWS:

# L TC USPRCADR

# L+1 CADR INTPRETX INTPRETX IS THE INTERPRETIVE CODING  
# RETURN IS TO L+2

USPRCADR TS LOC # SAVE A

CA BIT8  
TS EDOP

CA BBANK  
TS BANKSET # USER'S BBANK TO BANKSET

INDEX Q

CA 0  
TS FBANK  
MASK LOW10

XCH Q # INTERPRETIVE BANK TO FBANK  
XCH LOC # YIELDS INTERPRETIVE RELATIVE ADDRESS  
TCF Q+10000 # INTERPRETIVE ADDRESS TO Q, FETCHING L+1  
# L+1 TO LOC, RETRIEVING ORIGINAL A

# THERE ARE FOUR POSSIBLE SETTINGS FOR CHANNEL 07. (CHANNEL 07 CONTAINS SUPERBANK SETTING.)

| # | SUPERBANK | SETTING | S-REG. VALUE | PSEUDO-FIXED<br>BANK NUMBERS | OCTAL PSEUDO<br>ADDRESSES |
|---|-----------|---------|--------------|------------------------------|---------------------------|
| # | -----     | -----   | -----        | -----                        | -----                     |

|   |             |     |             |         |                 |                                                                                             |
|---|-------------|-----|-------------|---------|-----------------|---------------------------------------------------------------------------------------------|
| # | SUPERBANK 3 | OXX | 2000 - 3777 | 30 - 37 | 70000 - 107777  | (WHERE XX CAN BE ANYTHING AND<br>WILL USUALLY BE SEEN AS 11)                                |
| # | SUPERBANK 4 | 100 | 2000 - 3777 | 40 - 47 | 110000 - 127777 | (AS FAR AS IT CAN BE SEEN,<br>ONLY BANKS 40-43 WILL EVER BE<br>AND ARE PRESENTLY AVAILABLE) |
| # | SUPERBANK 5 | 101 | 2000 - 3777 | 50 - 57 | 130000 - 147777 | (PRESENTLY NOT AVAILABLE TO<br>THE USER)                                                    |
| # | SUPERBANK 6 | 110 | 2000 - 3777 | 60 - 67 | 150000 - 167777 | (PRESENTLY NOT AVAILABLE TO<br>THE USER)                                                    |

# \*\*\* THIS ROUTINE MAYBE CALLED BY ANY PROGRAM LOCATED IN BANKS 00 - 27. I.E., NO PROGRAM LIVING IN ANY  
# SUPERBANK SHOULD USE SUPERSW. \*\*\*

# SUPERSW MAYBE CALLED IN THIS FASHION:

|   |     |         |                                      |       |             |
|---|-----|---------|--------------------------------------|-------|-------------|
| # | CAF | ABBCON  | WHERE -- ABBCON                      | BBCON | SOMETHIN -- |
| # | TCR | SUPERSW | (THE SUPERBNK BITS ARE IN THE BBCON) |       |             |

# ...  
# .

# OR IN THIS FASHION :

|   |     |          |                                             |
|---|-----|----------|---------------------------------------------|
| # | CAF | SUPERSET | WHERE SUPERSET IS ONE OF THE FOUR AVAILABLE |
| # | TCR | SUPERSW  | SUPERBANK BIT CONSTANTS:                    |

|   |     |     |                    |
|---|-----|-----|--------------------|
| # | ... | ... | SUPER011 OCTAL 60  |
| # | .   | .   | SUPER100 OCTAL 100 |
| # | .   | .   | SUPER101 OCTAL 120 |
| # | .   | .   | SUPER110 OCTAL 140 |

|         |        |          |  |                                            |
|---------|--------|----------|--|--------------------------------------------|
| SUPERSW | EXTEND |          |  | # WRITE BITS 7-6-5 OF THE ACCUMULATOR INTO |
|         | WRITE  | SUPERBNK |  | # CHANNEL 07                               |
|         | TC     | Q        |  | # TC TO INSTRUCTION FOLLOWING              |
|         |        |          |  | # TC SUPERSW                               |

# INTERPRETER

# SECTION 1: DISPATCHER  
#

# ENTRY TO THE INTERPRETER. INTPRET SETS LOC TO THE FIRST INSTRUCTION, BANKSET TO THE BBANK OF THE  
# OBJECT INTERPRETIVE PROGRAM, AND INTBIT15 TO THE BIT15 CONTENTS OF FBANK. INTERPRETIVE PROGRAMS MAY BE IN  
# VIRTUALLY ALL BANKS PRESENT UNDER ANY SUPER-BANK SETTING, WITH THE RESTRICTION THAT PROGRAMS IN HIGH BANKS  
# (BIT15 OF FBANK = 1) DO NOT REFER TO LOWBANKS, AND VICE-VERSA. THE INTERPRETER DOES NOT SWITCH SUPERBANKS.  
# E-BANK SWITCHING OCCURS WHENEVER GENERAL ERASABLE (100-3777) IS ADDRESSED.

BLOCK 03

COUNT\* \$\$/INTER

INTPRET

RELINT  
EXTEND  
QXCH LOC

# SET LOC TO THE WORD FOLLOWING THE TC.

+2

CA BBANK  
TS BANKSET  
MASK BIT15  
TS INTBIT15

# INTERPRETIVE BRANCHES FINISH HERE.

# GET 15TH BIT FOR INDEXABLE ADDRESSES.

TS EDOP

# MAKE SURE NO INSTRUCTIONS LEFT OVER

TCF NEWOPS

# PICK UP OP CODE PAIR AND BEGIN.

INTRSM

LXCH BBANK  
TCF INTERPRET +3

# RESUME SUSPENDED INTERPRETIVE JOB

# DLOAD LOADS MPAC, MPAC +1, LEAVING ZERO IN MPAC +2.

DLOAD

EXTEND  
INDEX ADDRWD  
DCA 0

# LOAD DP C(C(ADDRWD)) INT MPAC,MPAC +1

SLOAD2

DXCH MPAC  
CAF ZERO

# ZERO MPAC +2

# AT THE END OF MOST INSTRUCTIONS, CONTROL IS GIVEN TO DANZIG TO DISPATCH THE NEXT OPERATION.

TS MPAC +2 # AND DECLARE DP MODE

NEWMODE TS MODE # PROLOGUE FOR MODE-CHANGING INSTRUCTIONS.

DANZIG CA BANKSET  
TS BBANK # SET BBANK BEFORE TESTING NEWJOB SO THAT  
# IT MAY BE SAVED DIRECTLY BY CHANJOB.

NOIBNKSW CCS EDOP  
TCF OPJUMP # SEE IF AN ORDER CODE IS LEFT OVER FROM  
# THE LAST PAIR RETRIEVED. IF SO, EXECUTE.  
# EDOP IS SET TO ZERO ON ITS RE-EDITIING.

CCS NEWJOB  
TCF CHANG2 # SEE IF A JOB OF HIGHER PRIORITY IS  
# PRESENT, AND IF SO, CHANGE JOBS.

INCR LOC # ADVANCE THE LOCATION COUNTER.

# ITRACE (1) REFERS TO "NEWOPS"  
NEWOPS INDEX LOC # ENTRY TO BEGIN BY PICKING OP CODE PAIR.  
CA 0 # MAY BE AN OPCODE PAIR OR A STORE CODE.  
CCS A # TEST SIGN AND GET DABS(A).  
TCF DOSTORE # PROCESS STORE CODE.

LOW7 OCT 177

TS EDOP  
MASK LOW7 # OP CODE PAIR. LEAVE THE OTHER IN EDOP  
# WHERE CCS EDOP WILL HONOR IT NEXT.

OPJUMP TS CYR  
CCS CYR  
TCF OPJUMP2 # LOWWD ENTERS HERE IF A RIGHT-HAND OP  
# CODE IS TO BE PROCESSED. TEST PREFICES.  
# TEST SECOND PREFIX BIT.

TCF EXIT # +0 OP CODE IS EXIT

# INTERPRETER

# PROCESS ADDRESSES WHICH MAY BE DIRECT, INDEXED, OR REFERENCE THE PUSHDOWN LIST.

|         |      |       |                                            |
|---------|------|-------|--------------------------------------------|
| ADDRESS | MASK | BIT1  | # SEE IF ADDRESS IS INDEXED. CYR CONTAINED |
|         | CCS  | A     | # 400XX, SO BIT 1 IS NOW AS IT WAS IN CYR. |
|         | TCF  | INDEX | # FORM INDEXED ADDRESS.                    |

|          |       |     |                                     |
|----------|-------|-----|-------------------------------------|
| DIRADRES | INDEX | LOC | # LOOK AHEAD TO NEXT WORD TO SEE IF |
| OCT40001 | CS    | 1   | # ADDRESS IS GIVEN.                 |

|     |        |
|-----|--------|
| CCS | A      |
| TCF | PUSHUP |

# IF NOT.

|      |     |    |
|------|-----|----|
| NEG4 | DEC | -4 |
|------|-----|----|

|      |        |                                       |
|------|--------|---------------------------------------|
| INCR | LOC    | # IF SO, TO SHOW WE PICKED UP A WORD. |
| TS   | ADDRWD |                                       |

# FINAL DIGESTION OF DIRECT ADDRESSES OF OP CODES WITH 01 PREFIX IS DONE HERE. IN EACH CASE, THE  
# REQUIRED 12-BIT SUB-ADDRESS IS LEFT IN ADDRWD, WITH ANY REQUIRED E OR F BANK SWITCHING DONE. ADDRESSES LESS  
# THAN 45D ARE TAKEN TO BE RELATIVE TO THE WORK AREA. THE OP CODE IS NOW IN BITS 1-5 OF CYR WITH BIT 14 = 1.

AD -ENDVAC # SEE IF ADDRESS RELATIVE TO WORK AREA.

CCS A  
AD -ENDERAS # IF NOT, SEE IF IN GENERAL ERASABLE.  
TCF IERASTST

NETZERO CA FIXLOC # IF SO, LEAVE THE MODIFIED ADDRESS IN  
ADS ADDRWD # ADDRWD AND DISPATCH.  
ITR15 INDEX CYR # THIS INDEX MAKES THE NEXT INSTRUCTION  
7 INDJUMP -1 # TCF INDJUMP + OP, EDITING CYR.

IERASTST EXTEND  
BZMF GEADDR # GO PROCESS GENERAL-ERASABLE ADDRESS.

MASK LOW10 # FIXED BANK ADDRESS. RESTORE AND ADD B15.  
AD LOW10 # SWITCH BANKS AND LEAVE SUBADDRESS IN  
XCH ADDRWD # ADDRWD FOR OPERAND RETRIEVAL. (THIS  
AD INTBIT15 # METHOD PRECLUDES USE OF THE LAST  
TS FBANK # LOCATION IN EACH FBANK.)  
ITR12 INDEX CYR

7 INDJUMP -1

GEADDR MASK LOW8  
AD OCT1400

XCH ADDRWD  
TS EBANK

ITR10 INDEX CYR  
7 INDJUMP -1

# THE FOLLOWING ROUTINE PROCESSES INTERPRETIVE INDEXED ADDRESSES. AN INTERPRETER INDEX REGISTER MAY  
# CONTAIN THE ADDRESS OF ANY ERASABLE REGISTER (0-42 BEING RELATIVE TO THE VAC AREA) OR ANY INTERPRETIVE PROGRAM  
# BANK, OR ANY INTEGER IN THAT RANGE.

DODLOAD\*      CAF      DLOAD\*      # STODL\* COMES HERE TO PROCESS LOAD ADR.  
TS      CYR      # (STOVL\* ENTERS HERE).

INDEX      CA      FIXLOC      # SET UP INDEX LOCATION.  
TS      INDEXLOC  
INCR      LOC      # (ADDRESS ALWAYS GIVEN).  
INDEX      LOC

CS      0  
CCS      A      # INDEX 2 IF ADDRESS STORED COMPLEMENTED.  
INCR      INDEXLOC  
NOOP

TS      ADDRWD      # 14 BIT ADDRESS TO ADDRWD.  
MASK      HIGH4      # IF ADDRESS GREATER THAN 2K, ADD INTBIT15  
EXTEND  
BZF      INDEX2  
CA      INTBIT15  
ADS      ADDRWD

INDEX2      INDEX      INDEXLOC  
CS      X1  
ADS      ADDRWD      # DO AUGMENT, IGNORING AND CORRECTING OVF.

MASK      HIGH9      # SEE IF ADDRESS IS IN WORK AREA.  
EXTEND

BZF      INDWORK  
MASK      HIGH4      # SEE IF IN FIXED BANK.  
EXTEND

BZF      INDERASE

CA      ADDRWD      # IN FIXED -- SWITCH BANKS AND CREATE  
TS      FBANK      # SUB-ADDRESS  
MASK      LOW10  
AD      2K

ITR11      TS      ADDRWD  
INDEX      CYR  
3      INDJUMP -1

INDWORK      CA      FIXLOC      # MAKE ADDRWD RELATIVE TO WORK AREA.  
TCF      ITR13      -1

INDERASE      CA      OCT1400  
XCH      ADDRWD  
TS      EBANK  
MASK      LOW8  
-1      ADS      ADDRWD

```
INDJUMP -1
```



# PUSH-UP ROUTINES. WHEN NO OPERAND ADDRESS IS GIVEN, THE APPROPRIATE OPERAND IS TAKEN FROM THE PUSH-DOWN  
# LIST. IN MOST CASES THE MODE OF THE RESULT (VECTOR OR SCALAR) OF THE LAST ARITHMETIC OPERATION PERFORMED  
# IS THE SAME AS THE TYPE OF OPERAND DESIRED (ALL ADD/SUBTRACT ETC.). EXCEPTIONS TO THIS GENERAL RULE ARE LISTED  
# BELOW (NOTE THAT IN EVERY CASE THE MODE REGISTER IS LEFT INTACT):

- # 1. VXSC AND V/SC WANT THE OPPOSITE TYPE OF OPERAND, E.G., IF THE LAST OPERATION YIELDED A VECTOR  
# RESULT, VXSC WANTS A SCALAR.
- # 2. THE LOAD CODES SHOULD LOAD THE ACCUMULATOR INDEPENDENT OF THE RESULT OF THE LAST OPERATION. THIS  
# INCLUDES VLOAD, DLOAD, TLOAD, PDDL, AND PDVL (NO PUSHUP WITH SLOAD).
- # 3. SOME ARITHMETIC OPERATIONS REQUIRE A STANDARD TYPE OF OPERAND REGARDLESS OF THE PREVIOUS OPERATION.  
# THIS INCLUDES SIGN WANTING DP AND TAD REQUIRING TP.

PUSHUP CAF OCT23 # IF THE LOW 5 BITS OF CYR ARE LESS THAN  
MASK CYR # 20, THIS OP REQUIRES SPECIAL ATTENTION.  
AD -OCT10 # (NO -0).

CCS A  
TCF REGUP # FOR ALL CODES GREATER THAN OCT 7.

-OCT10 OCT -10

AD NEG4 # WE NOW HAVE 7 -- OP CODE (MOD4). SEE IF  
CCS A # THE OP CODE (MOD4) IS THREE (REVERSE).  
INDEX A # NO -- THE MODE IS DEFINITE. PICK UP THE  
CS NO.WDS  
TCF REGUP +2

INDEX MODE # FOR VXSC AND V/SC WE WANT THE REQUIRED  
CS REVCNT # PUSHLOC DECREMENT WITHOUT CHANGING THE  
TCF REGUP +2 # MODE AT THIS TIME.

REGUP INDEX MODE # MOST ALL OP CODES PUSHUP HERE.

+2 CS NO.WDS  
ADS PUSHLOC

TS ADDRWD  
ITR14 INDEX CYR  
7 INDJUMP -1 # (THE INDEX MAKES THIS A TCF.)

OCT 2 # REVERSE PUSHUP DECREMENT. VECTOR TAKES 2  
REVCNT OCT 6 # WORDS, SCALAR TAKES 6.

OCT 6  
NO.WDS OCT 2 # CONVENTIONAL DECREMENT IS 6 WORDS VECTOR  
OCTAL3 OCT 3 # 2 IN DP, AND 3 IN TP.  
OCT 6

# INTERPRETER

# TEST THE SECOND PREFIX BIT TO SEE IF THIS IS A MISCELLANEOUS OR A UNARY/SHORT SHIFT OPERATION.

|         |     |         |                                           |
|---------|-----|---------|-------------------------------------------|
| OPJUMP2 | CCS | CYR     | # TEST SECOND PREFIX BIT.                 |
|         | TCF | OPJUMP3 | # TEST THIRD BIT TO SEE IF UNARY OR SHIFT |

-ENDVAC            DEC            -45

# THE FOLLOWING ROUTINE PROCESSES ADDRESSES OF SUFFIX CLASS 10. THEY ARE BASICALLY WORK AREA ADDRESSES  
# IN THE RANGE 0-52, ERASABLE ECADR CONSTANTS FROM 100-3777, AND FCADRS ABOVE THAT. ALL 15 BITS ARE AVAILABLE  
# IN CONTRAST TO SUFFIX 1, IN WHICH ONLY THE LOW ORDER 14 ARE AVAILABLE.

|          |       |          |                                     |
|----------|-------|----------|-------------------------------------|
| 15BITADR | INCR  | LOC      | # (ENTRY HERE FROM STCALL).         |
|          | INDEX | LOC      | # PICK UP ADDRESS WORD.             |
|          | CA    | 0        |                                     |
|          | TS    | POLISH   | # WE MAY NEED A SUBADDRESS LATER.   |
|          | CAF   | LOW7+2K  | # THESE INSTRUCTIONS ARE IN BANK 1. |
|          | TS    | FBANK    |                                     |
| ITR7     | MASK  | CYR      |                                     |
|          | INDEX | A        |                                     |
|          | TCF   | MISCJUMP |                                     |

1412THE  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
1

```
COMPLETE THE DISPATCHING OF UNARY AND SHORT SHIFT OPERATIONS.

OPJUMP3 TS FBANK # CALL IN BANK 0 (BIT5S 11-15 OF A ARE 0.)
ITRACE (6) REFERS TO "OPJUMP3"
 CCS CYR # TEST THIRD PREFIX BIT.
 INDEX A # THE DECREMENTED UNARY CODE IS IN BITS
 TCF UNAJUMP # 1-4 OF A (ZERO, EXIT, HAS BEEN DETECTED)

 CCS MODE # IT'S A SHORT SHIFT CODE. SEE IF PRESENT
 TCF SHORTT # SCALAR OR VECTOR.
 TCF SHORTT
 TCF SHORTTV # CALLS THE APPROPRIATE ROUTINE.

FBANKMSK EQUALS BANKMASK
LVBUF ADRES VBUF
```

# THE FOLLOWING IS THE JUMP TABLE FOR OP CODES WHICH MAY HAVE INDEXABLE ADDRESSES OR MAY PUSH UP.

|         |     |        |                                             |
|---------|-----|--------|---------------------------------------------|
| INDJUMP | TCF | VLOAD  | # 00 -- LOAD MPAC WITH A VECTOR.            |
|         | TCF | TAD    | # 01 -- TRIPLE PRECISION ADD TO MPAC.       |
|         | TCF | SIGN   | # 02 -- COMPLEMENT MPAC (V OR SC) IF X NEG. |
|         | TCF | VXSC   | # 03 -- VECTOR TIMES SCALAR.                |
|         | TCF | CGOTO  | # 04 -- COMPUTED GO TO.                     |
|         | TCF | TLOAD  | # 05 -- LOAD MPAC WITH TRIPLE PRECISION.    |
|         | TCF | DLOAD  | # 06 -- LOAD MPAC WITH A DP SCALAR.         |
|         | TCF | V/SC   | # 07 -- VECTOR DIVIDED BY A SCALAR.         |
|         | TCF | SLOAD  | # 10 -- LOAD MPACIN SINGLE PRECISION.       |
|         | TCF | SSP    | # 11 -- SET SINGLE PRECISION INTO X.        |
|         | TCF | PDDL   | # 12 -- PUSH DOWN MPAC AND RE-LOAD IN DP.   |
|         | TCF | MXV    | # 13 -- MATRIX POST-MULTIPLIED BY VECTOR.   |
|         | TCF | PDVL   | # 14 -- PUSH DOWN AND VECTOR LOAD.          |
|         | TCF | CCALL  | # 15 -- COMPUTED CALL.                      |
|         | TCF | VXM    | # 16 -- MATRIX PRE-MULTIPLIED BY VECTOR.    |
|         | TCF | TSLC   | # 17 -- NORMALIZE MPAC (SCALAR ONLY).       |
|         | TCF | DMPR   | # 20 -- DP MULTIPLY AND ROUND.              |
|         | TCF | DDV    | # 21 -- DP DIVIDE BY.                       |
|         | TCF | BDDV   | # 22 -- DP DIVIDE INTO.                     |
|         | TCF | GSHIFT | # 23 -- GENERAL SHIFT INSTRUCTION           |
|         | TCF | VAD    | # 24 -- VECTOR ADD.                         |
|         | TCF | VSU    | # 25 -- VECTOR SUBTRACT.                    |
|         | TCF | BVSU   | # 26 -- VECTOR SUBTRACT FROM.               |
|         | TCF | DOT    | # 27 -- VECTOR DOT PRODUCT.                 |
|         | TCF | VXV    | # 30 -- VECTOR CROSS PRODUCT.               |
|         | TCF | VPROJ  | # 31 -- VECTOR PROJECTION.                  |
|         | TCF | DSU    | # 32 -- DP SUBTRACT.                        |
|         | TCF | BDSU   | # 33 -- DP SUBTRACT FROM.                   |
|         | TCF | DAD    | # 34 -- DP ADD.                             |
|         | TCF |        | # 35 -- AVAILABLE                           |
|         | TCF | DMP1   | # 36 -- DP MULTIPLY.                        |
|         | TCF | SETPD  | # 37 -- SET PUSH DOWN POINTER (DIRECT ONLY) |

# CODES 10 AND 14 MUST NOT PUSH UP. CODE 04 MAY BE USED FOR VECTOR DECLARE BEFORE PUSHUP IF DESIRED.

# INTERPRETER

# THE FOLLOWING JUMP TABLE APPLIES TO INDEX, BRANCH, AND MISCELLANEOUS INSTRUCTIONS.

|          |     |          |                                             |
|----------|-----|----------|---------------------------------------------|
| MISCJUMP | TCF | AXT      | # 00 -- ADDRESS TO INDEX TRUE.              |
|          | TCF | AXC      | # 01 -- ADDRESS TO INDEX COMPLEMENTED.      |
|          | TCF | LXA      | # 02 -- LOAD INDEX FROM ERASABLE.           |
|          | TCF | LXC      | # 03 -- LOAD INDEX FROM COMPLEMENT OF ERAS. |
|          | TCF | SXA      | # 04 -- STORE INDEX IN ERASABLE.            |
|          | TCF | XCHX     | # 05 -- EXCHANGE INDEX WITH ERASABLE.       |
|          | TCF | INCR     | # 06 -- INCREMENT INDEX REGISTER.           |
|          | TCF | TIX      | # 07 -- TRANSFER ON INDEX.                  |
|          | TCF | XAD      | # 10 -- INDEX REGISTER ADD FROM ERASABLE.   |
|          | TCF | XSU      | # 11 -- INDEX SUBTRACT FROM ERASABLE.       |
|          | TCF | BZE/GOTO | # 12 -- BRANCH ZERO AND GOTO                |
|          | TCF | BPL/BMN  | # 13 -- BRANCH PLUS AND BRANCH MINUS.       |
|          | TCF | RTB/BHIZ | # 14 -- RETURN TO BASIC AND BRANCH HI ZERO. |
|          | TCF | CALL/ITA | # 15 -- CALL AND STORE QPRET.               |
|          | TCF | SW/      | # 16 -- SWITCH INSTRUCTIONS AND AVAILABLE.  |
|          | TCF | BOV(B)   | # 17 -- BRANCH ON OVERFLOW TO BASIC OR INT. |

1412THE

# THE FOLLOWING JUMP TABLE APPLIES TO UNARY INSTRUCTIONS.

|         |        |            |                                            |
|---------|--------|------------|--------------------------------------------|
| UNAJUMP | COUNT* | \$\$/INTER |                                            |
|         | BANK   | 0          | # 00 -- EXIT -- DETECTED EARLIER.          |
|         | TCF    | SQRT       | # 01 -- SQUARE ROOT.                       |
|         | TCF    | SINE       | # 02 -- SIN.                               |
|         | TCF    | COSINE     | # 03 -- COS.                               |
|         | TCF    | ARCSIN     | # 04 -- ARC SIN.                           |
|         | TCF    | ARCCOS     | # 05 -- ARC COS.                           |
|         | TCF    | DSQ        | # 06 -- DP SQUARE.                         |
|         | TCF    | ROUND      | # 07 -- ROUND TO DP.                       |
|         | TCF    | COMP       | # 10 -- COMPLEMENT VECTOR OR SCALAR        |
|         | TCF    | VDEF       | # 11 -- VECTOR DEFINE.                     |
|         | TCF    | UNIT       | # 12 -- UNIT VECTOR.                       |
|         | TCF    | ABVALABS   | # 13 -- LENGTH OF VECTOR OR MAG OF SCALAR. |
|         | TCF    | VSQ        | # 14 -- SQUARE OF LENGTH OF VECTOR.        |
|         | TCF    | STADR      | # 15 -- PUSH UP ON STORE CODE.             |
|         | TCF    | RVQ        | # 16 -- RETURN VIA QPRET.                  |
|         | TCF    | PUSH       | # 17 -- PUSH MPAC DOWN.                    |

# SECTION 2      LOAD AND STORE PACKAGE.

#  
# A SET OF EIGHT STORE CODES IS PROVIDED AS THE PRIMARY METHOD OF STORING THE MULTI-PURPOSE  
# ACCUMULATOR (MPAC). IF IN THE DANZIG SECTION LOC REFERS TO AN ALGEBRAICALLY POSITIVE WORD, IT IS TAKEN AS A  
# STORE CODE WITH A CORRESPONDING ERASABLE ADDRESS. MOST OF THESE CODES ARE TWO ADDRESS, SPECIFYING THAT THE WORD  
# FOLLOWING THE STORE CODE IS TO BE USED AS AN ADDRESS FROM WHICH TO RE-LOAD MPAC. FOUR OPTIONS ARE AVAILABLE:  
#  
#      1. STORE            STORE MPAC. THE E ADDRESS MAY BE INDEXED.  
#      2. STODL           STORE MPAC AND RE-LOAD IT IN DP WITH THE NEXT ADDRESS (THE LOAD MAY BE INDEXED).  
#      3. STOVL           STORE MPAC AND RE-LOAD A VECTOR (AS ABOVE).  
#      4. STCALL          STORE AND DO A CALL (BOTH ADDRESSES MUST BE DIRECT HERE).

# STODL AND STOVL WILL TAKE FROM THE PUSH-DOWN LIST IF NO LOAD ADDRESS IS GIVEN.

BLOCK    3

COUNT\*    \$\$/INTER

STADR      CA      BANKSET      # THE STADR CODE (PUSHUP UP ON STORE  
            TS      FBANK      # ADDRESS) ENTERS HERE.

            INCR    LOC

ITR1       INDEX   LOC      # THE STORECODE WAS STORED COMPLEMENTED TO  
            CS      0        # MAKE IT LOOK LIKE AN OPCODE PAIR.  
            AD      NEGONE    # (YUL CAN'T REMOVE 1 BECAUSE OF EARLY CCS)

DOSTORE    TS      ADDRWD      # ENTRY FROM DISPATCHER. SAVE THE ERASABLE  
            MASK    LOW11      # ADDRESS AND JUMP ON THE STORE CODE NO.

            XCH      ADDRWD

            MASK    B12T14

            EXTEND

            MP      BIT5

            INDEX   A

            TCF     STORJUMP

# INTERPRETER

# STORE CODE JUMP TABLE. CALLS THE APPROPRIATE STORING ROUTINE AND EXITS TO DANZIG OR TO ADDRESS WITH  
# A SUPPLIED OPERATION CODE.

#  
# STORE STORE,1 AND STORE,2 RETURN TO DANZIG, THUS RESETTING THE EBANK TO ITS STATE AT INTPRET.

|          |     |         |                           |
|----------|-----|---------|---------------------------|
| STORJUMP | TC  | STORE   | # STORE.                  |
|          | TCF | DANZIG  | # PICK UP NEW OP CODE(S). |
|          | TC  | STORE,1 |                           |
|          | TCF | DANZIG  |                           |
|          | TC  | STORE,2 |                           |
|          | TCF | DANZIG  |                           |

|  |     |         |          |
|--|-----|---------|----------|
|  | TC  | STORE   | # STODL. |
|  | TCF | DODLOAD |          |

|  |     |          |                                    |
|--|-----|----------|------------------------------------|
|  | TC  | STORE    | # STODL WITH INDEXED LOAD ADDRESS. |
|  | TCF | DODLOAD* |                                    |

|  |     |         |          |
|--|-----|---------|----------|
|  | TC  | STORE   | # STOVL. |
|  | TCF | DOVLOAD |          |

|  |     |          |                                    |
|--|-----|----------|------------------------------------|
|  | TC  | STORE    | # STOVL WITH INDEXED LOAD ADDRESS. |
|  | TCF | DOVLOAD* |                                    |

|  |     |          |                         |
|--|-----|----------|-------------------------|
|  | TC  | STORE    | # STOTC.                |
|  | CAF | CALLCODE |                         |
|  | TS  | CYR      |                         |
|  | TCF | 15BITADR | # GET A 15 BIT ADDRESS. |



# INTERPRETER

# STORE CODE ADDRESS PROCESSOR.

STORE,1            INDEX    FIXLOC  
                  CS        X1  
                  TCF        PRESTORE

STORE,2            INDEX    FIXLOC  
                  CS        X2

PRESTORE          ADS       ADDRWD            # RESULTANT ADDRESS IS IN ERASABLE.

STORE             CS        ADDRWD

AD        DEC45  
CCS       A            # DOES THE ADDRESS POINT TO THE WORK AREA?  
CA        FIXLOC       # YES.

TCF       AHEAD5  
CA        OCT1400       # NO.    SET EBANK & MAKE UP SUBADDRESS.  
XCH       ADDRWD

AHEAD5            TS        EBANK  
                  MASK     LOW8  
                  ADS       ADDRWD

1412THE



# INTERPRETER

# STORING ROUTINES. STORE DP, TP, OR VECTOR AS INDICATED BY MODE.

STARTSTO            EXTEND            # MPAC,+1 MUST BE STORED IN ANY EVENT.  
# ITRACE (5) REFERS TO "STARTSTO".

DCA        MPAC  
INDEX      ADDRWD  
DXCH       0

CCS        MODE  
TCF        TSTORE  
TC          Q

VSTORE            EXTEND            +3  
DCA        MPAC

INDEX      ADDRWD  
DXCH       2

EXTEND            +5  
DCA        MPAC

INDEX      ADDRWD  
DXCH       4  
TC          Q

TSTORE            CA        MPAC        +2  
INDEX      ADDRWD

TS          2  
TC          Q

1412THE

# ROUTINES TO BEGIN PROCESSING OF THE SECOND ADDRES ASSOCIATED WITH ALL STORE-TYPE CODES EXCEPT STORE  
# ITSELF.

DODLOAD            CAF        DLOADCOD  
                     TS        CYR  
                     TCF        DIRADRES        # GO GET A DIRECT ADDRESS.

DOVLOAD            CAF        VLOADCOD  
                     TS        CYR  
                     TCF        DIRADRES

DOVLOAD\*           CAF        VLOAD\*  
                     TCF        DODLOAD\* +1        # PROLOGUE TO INDEX ROUTINE.

# THE FOLLOWING LOAD INSTRUCTIONS ARE PROVIDED FOR LOADING THE MULTI-PURPOSE ACCUMULATOR MPAC.

TLOAD INDEX ADDRWD  
CA 2 # LOAD A TRIPLE PRECISION ARGUMENT INTO  
TS MPAC +2 # THE FIRST THREE MPAC REGISTERS, WITH THE  
# CONTENTS OF THE OTHER FOUR IRRELEVANT.

EXTEND  
INDEX ADDRWD  
DCA 0  
TMODE DXCH MPAC  
CAF ONE  
TCF NEWMODE # DECLARE TRIPLE PRECISION MODE.

SLOAD ZL # LOAD A SINGLE PRECISION NUMBER INTO  
INDEX ADDRWD # MPAC, SETTING MPAC+1,2 TO ZERO. THE  
CA 0 # CONTENTS OF THE REMAINING MPAC REGISTERS  
TCF SLOAD2 # ARE IRRELEVANT.

VLOAD EXTEND # LOAD A DOUBLE PRECISION VECTOR INTO  
INDEX ADDRWD # MPAC,+1, MPAC+3,4, AND MPAC+5,6. THE  
DCA 0 # CONTENTS OF MPAC +2 ARE IRRELEVANT.  
DXCH MPAC

ENDVLOAD EXTEND # PDVL COMES HERE TO FINISH UP FOR DP, TP.

INDEX ADDRWD  
DCA 2  
DXCH MPAC +3

+4 EXTEND # TPDVL FINISHES HERE.

INDEX ADDRWD  
DCA 4  
DXCH MPAC +5

VMODE CS ONE # DECLARE VECTOR MODE.  
TCF NEWMODE

# INTERPRETER

# THE FOLLOWING INSTRUCTIONS ARE PROVIDED FOR STORING OPERANDS IN THE PUSHDOWN LIST:

- # 1. PUSH PUSHDOWN AND NO LOAD.
- # 2. PDDL PUSHDOWN AND DOUBLE PRECISION LOAD.
- # 3. PDVL PUSHDOWN AND VECTOR LOAD.

PDDL EXTEND INDEX ADDRWD # LOAD MPAC,+1, PUSHING THE FORMER  
DCA 0 # CONTENTS DOWN.

DXCH MPAC  
INDEX PUSHLOC  
DXCH 0

INDEX MODE # ADVANCE THE PUSHDOWN POINTER APPRO-  
CAF NO.WDS # PRIATELY.  
ADS PUSHLOC

CCS MODE  
TCF ENDTYPUSH  
TCF ENDDPUSH

ENDVPUSH TS MODE # NOW DP.  
TS MPAC +2  
DXCH MPAC +3 # PUSH DOWN THE REST OF THE VECTOR HERE.

INDEX PUSHLOC  
DXCH 0 -4

DXCH MPAC +5  
INDEX PUSHLOC  
DXCH 0 -2

TCF DANZIG

ENDDPUSH TS MPAC +2 # SET MPAC +2 TO ZERO AND EXIT ON DP.  
TCF DANZIG

ENDTPUSH TS MODE  
XCH MPAC +2 # ON TRIPLE, SET MPAC +2 TO ZERO, PUSHING  
INDEX PUSHLOC # DOWN THE OLD CONTENTS

+2 TS 0 -1  
TCF DANZIG

## # PDVL -- PUSHDOWN AND VECTOR LOAD

PDVL                    EXTEND                    # RELOAD MPAC AND PUSH DOWN ITS CONTENTS.

INDEX           ADDRWD

DCA             0

DXCH            MPAC

INDEX           PUSHLOC

DXCH            0

INDEX           MODE

CAF            NO.WDS

ADS            PUSHLOC

CCS            MODE

TCF            TPDVL

TCF            ENDVLOAD

# ADVANCE THE PUSHDOWN POINTER.

# TEST PAST MODE.

# JUST LOAD LAST FOUR REGISTERS ON DP.

VPDVL                   EXTEND                    # PUSHDOWN AND RE-LOAD LAST TWO COMPONENTS

INDEX           ADDRWD

DCA             2

DXCH            MPAC       +3

INDEX           PUSHLOC

DXCH            0        -4

EXTEND

INDEX           ADDRWD

DCA             4

DXCH            MPAC       +5

INDEX           PUSHLOC

DXCH            0        -2

TCF            DANZIG

TPDVL                   EXTEND                    # ON TP, WE MUST LOAD THE Y COMPONENT  
INDEX           ADDRWD                    # BEFORE STORING MPAC +2 IN CASE THIS IS A  
DCA             2                         # PUSHUP.

DXCH            MPAC       +3

CA             MPAC       +2

INDEX           PUSHLOC                    # IN DP.

TS             0        -1

TCF            ENDVLOAD +4

# SSP (STORE SINGLE PRECISION) IS EXECUTED HERE.

SSP                    INCR            LOC                    # PICK UP THE WORD FOLLOWING THE GIVEN  
INDEX           LOC                        # ADDRESS AND STORE IT AT X.STORE1                CA             0  
INDEX           ADDRWD                    # SOME INDEX AND MISCELLANEOUS OPS END  
TS             0                            # HERE.

| # INTERPRETER |            | PAGE 1022 |    |
|---------------|------------|-----------|----|
| 1             | TCF DANZIG |           | 1  |
| 2             |            |           | 2  |
| 3             |            |           | 3  |
| 4             |            |           | 4  |
| 5             |            |           | 5  |
| 6             |            |           | 6  |
| 7             |            |           | 7  |
| 8             |            |           | 8  |
| 9             |            |           | 9  |
| 10            |            |           | 10 |
| 11            |            |           | 11 |
| 12            |            |           | 12 |
| 13            |            |           | 13 |
| 14            |            |           | 14 |
| 15            |            |           | 15 |
| 16            |            |           | 16 |
| 17            |            |           | 17 |
| 18            |            |           | 18 |
| 19            |            |           | 19 |
| 20            |            |           | 20 |
| 21            |            |           | 21 |
| 22            |            |           | 22 |
| 23            |            |           | 23 |
| 24            |            |           | 24 |
| 25            |            |           | 25 |
| 26            |            |           | 26 |
| 27            |            |           | 27 |
| 28            |            |           | 28 |
| 29            |            |           | 29 |
| 30            |            |           | 30 |
| 31            |            |           | 31 |
| 32            |            |           | 32 |
| 33            |            |           | 33 |
| 34            |            |           | 34 |
| 35            |            |           | 35 |
| 36            |            |           | 36 |
| 37            |            |           | 37 |
| 38            |            |           | 38 |
| 39            |            |           | 39 |
| 40            |            |           | 40 |
| 41            |            |           | 41 |
| 42            |            |           | 42 |
| 43            |            |           | 43 |
| 44            |            |           | 44 |
| 45            |            |           | 45 |
| 46            |            |           | 46 |
| 47            |            |           | 47 |
| 48            |            |           | 48 |
| 49            |            |           | 49 |
| 50            |            |           | 50 |
| 51            |            |           | 51 |
| 52            |            |           | 52 |
| 53            |            |           | 53 |
| 54            |            |           | 54 |
| 55            |            |           | 55 |
| 56            |            |           | 56 |
| 57            |            |           | 57 |
| 58            |            |           | 58 |
| 59            |            |           | 59 |
| 60            |            |           | 60 |

1412THE

# SEQUENCE CHANGING AND SUBROUTINE CALLING OPTIONS.

#

# THE FOLLOWING OPERATIONS ARE AVAILABLE FOR SEQUENCING CHANGING, BRANCHING, AND CALLING SUBROUTINES:

|   |    |       |                                   |
|---|----|-------|-----------------------------------|
| # | 1. | GOTO  | GO TO.                            |
| # | 2. | CALL  | CALL SUBROUTINE SETTING QPRET.    |
| # | 3. | CGOTO | COMPUTED GO TO.                   |
| # | 4. | CCALL | COMPUTED CALL.                    |
| # | 7. | BPL   | BRANCH IF MPAC POSITIVE OR ZERO.  |
| # | 8. | BZE   | BRANCH IF MPAC ZERO.              |
| # | 9. | BMN   | BRANCH IF MPAC NEGATIVE NON-ZERO. |

CCALL            INCR    LOC            # MAINTAIN LOC FOR QPRET COMPUTATION

INDEX  
CAFLOC  
0

# GET BASE ADDRESS OF CADR LIST.

INDEX  
AD  
TSADDRWD  
0  
FBANK

# ADD INCREMENT.

# SELECT DESIRED CADR.

MASK  
INDEX  
CAFLOW10  
A  
10000

TS    POLISH

CALL

CA

BANKSET

# FOR ANY OF THE CALL OPTIONS, MAKE UP THE

MASK

BANKMASK

# ADDRESS OF THE NEXT OP-CODE PAIR/STORE

AD

BANKMASK

# CODE AND LEAVE IT IN QPRET. NOTE THAT

AD

LOC

# BANKMASK = -(2000 - 1).

INDEX

FIXLOC

TS

QPRET

GOTO

+1

CA

POLISH

# BASIC BRANCHING SEQUENCE.

MASK

HIGH4

EXTEND

BZF

GOTOERS

# SEE IF ADDRESS POINTS TO FIXED OR ERAS.

CA

BANKSET

# SET EBANK PART OF BBANK. NEXT, SET UP

TS

BBANK

# FBANK. THE COMBINATION IS PICKED UP &amp;

CA

POLISH

# PUT INTO BANKSET AT INTPRET +2.

TS

FBANK

MASK

LOW10

AD

2K

TS

LOC

TCF

INTPRET +3

GOTOERS

CA

POLISH

# THE GIVEN ADDRESS IS IN ERASABLE -- SEE

AD

-ENDVAC

# IF RELATIVE TO THE WORK ARA.

CCS

A

CA

POLISH

# GENERAL ERASABLE.

TCF

GOTOGE



SWBRANCH

|    |                                                                                    |       |                               |                                            |    |
|----|------------------------------------------------------------------------------------|-------|-------------------------------|--------------------------------------------|----|
| 1  | # TRIPLE PRECISION BRANCHING ROUTINE. IF CALLING TC IS AT L, RETURN IS AS FOLLOWS: |       |                               |                                            | 2  |
| 2  | #                                                                                  | L+1   | IF MPAC IS GREATER THAN ZERO. |                                            | 3  |
| 3  | #                                                                                  | L+2   | IF MPAC IS EQUAL TO +0 OR -0. |                                            | 4  |
| 4  | #                                                                                  | L+3   | IF MPAC IS LESS THAN ZERO.    |                                            | 5  |
| 5  |                                                                                    |       |                               |                                            | 6  |
| 6  |                                                                                    |       |                               |                                            | 7  |
| 7  | BRANCH                                                                             | CCS   | MPAC                          |                                            | 8  |
| 8  |                                                                                    | TC    | Q                             |                                            | 9  |
| 9  |                                                                                    | TCF   | +2                            | # ON ZERO.                                 | 10 |
| 10 |                                                                                    | TCF   | NEG                           |                                            | 11 |
| 11 |                                                                                    |       |                               |                                            | 12 |
| 12 |                                                                                    | CCS   | MPAC                          | +1                                         | 13 |
| 13 |                                                                                    | TC    | Q                             |                                            | 14 |
| 14 |                                                                                    | TCF   | +2                            |                                            | 15 |
| 15 |                                                                                    | TCF   | NEG                           |                                            | 16 |
| 16 |                                                                                    |       |                               |                                            | 17 |
| 17 |                                                                                    | CCS   | MPAC                          | +2                                         | 18 |
| 18 |                                                                                    | TC    | Q                             |                                            | 19 |
| 19 |                                                                                    | TCF   | +2                            |                                            | 20 |
| 20 |                                                                                    | TCF   | NEG                           |                                            | 21 |
| 21 |                                                                                    |       |                               |                                            | 22 |
| 22 | Q+1                                                                                | INDEX | Q                             |                                            | 23 |
| 23 |                                                                                    | TC    | 1                             |                                            | 24 |
| 24 |                                                                                    |       |                               |                                            | 25 |
| 25 | NEG                                                                                | INDEX | Q                             | # IF FIRST NON-ZERO REGISTER WAS NEGATIVE. | 26 |
| 26 |                                                                                    | TC    | 2                             |                                            | 27 |
| 27 |                                                                                    |       |                               |                                            | 28 |
| 28 | Q+2                                                                                | =     | NEG                           |                                            | 29 |
| 29 |                                                                                    |       |                               |                                            | 30 |
| 30 | # ITRACE (3) REFERS TO "EXIT".                                                     |       |                               |                                            | 31 |
| 31 |                                                                                    |       |                               |                                            | 32 |
| 32 | EXIT                                                                               | CA    | BANKSET                       | # RESTORE USER'S BANK SETTING, AND LEAVE   | 33 |
| 33 |                                                                                    | TS    | BBANK                         | # INTERPRETIVE MODE.                       | 34 |
| 34 |                                                                                    | INDEX | LOC                           |                                            | 35 |
| 35 |                                                                                    | TC    | 1                             |                                            | 36 |
| 36 |                                                                                    |       |                               |                                            | 37 |
| 37 |                                                                                    |       |                               |                                            | 38 |
| 38 |                                                                                    |       |                               |                                            | 39 |
| 39 |                                                                                    |       |                               |                                            | 40 |
| 40 |                                                                                    |       |                               |                                            | 41 |
| 41 |                                                                                    |       |                               |                                            | 42 |
| 42 |                                                                                    |       |                               |                                            | 43 |
| 43 |                                                                                    |       |                               |                                            | 44 |
| 44 |                                                                                    |       |                               |                                            | 45 |
| 45 |                                                                                    |       |                               |                                            | 46 |
| 46 |                                                                                    |       |                               |                                            | 47 |
| 47 |                                                                                    |       |                               |                                            | 48 |
| 48 |                                                                                    |       |                               |                                            | 49 |
| 49 |                                                                                    |       |                               |                                            | 50 |
| 50 |                                                                                    |       |                               |                                            | 51 |
| 51 |                                                                                    |       |                               |                                            | 52 |
| 52 |                                                                                    |       |                               |                                            | 53 |
| 53 |                                                                                    |       |                               |                                            | 54 |
| 54 |                                                                                    |       |                               |                                            | 55 |
| 55 |                                                                                    |       |                               |                                            | 56 |
| 56 |                                                                                    |       |                               |                                            | 57 |
| 57 |                                                                                    |       |                               |                                            | 58 |
| 58 |                                                                                    |       |                               |                                            | 59 |
| 59 |                                                                                    |       |                               |                                            | 60 |
| 60 |                                                                                    |       |                               |                                            | 61 |

# SECTION 3 -- ADD/SUBTRACT PACKAGE.

#

# THE FOLLOWING OPERATIONS ARE PROVIDED FOR ADDING TO AND SUBTRACTING FROM THE MULTI-PURPOSE ACCUMULATOR

# MPAC:

# 1. DAD DOUBLE PRECISION ADD.

# 2. DSU DOUBLE PRECISION SUBTRACT.

# 3. BDSU DOUBLE PRECISION SUBTRACT FROM.

# 4. TAD TRIPLE PRECISION ADD.

# 5. VAD VECTOR ADD.

# 6. VSU VECTOR SUBTRACT.

# 7. BVSU VECTOR SUBTRACT FROM.

# THE INTERPRETIVE OVERFLOW INDICATOR OVFind IS SET NON-ZERO IF OVERFLOW OCCURS IN ANY OF THE ABOVE.

VSU CAF BIT15 # CHANGES 0 TO DCS.

TCF +2

VAD CAF PRI030 # CHANGES 0 TO DCA.

ADS ADDRWD

EXTEND

INDEX ADDRWD

READ HISCALAR # DCA 2 OR DCS 2

DAS MPAC +3

EXTEND # CHECK OVERFLOW.

BZF +2

TC OVERFLWY

EXTEND

INDEX ADDRWD

READ CHAN5 # DCA 4 OR DCS 4

DAS MPAC +5

EXTEND

BZF +2

TC OVERFLWZ

EXTEND

INDEX ADDRWD

READ LCHAN # DCA 0 OR DCS 0

TCF ENDVXV

DAD EXTEND

INDEX ADDRWD

ENDVXV DCA 0

DAS MPAC

EXTEND

BZF DANZIG

# VXV FINISHES HERE.

| # INTERPRETER |        |           | PAGE 1027          |    |    |
|---------------|--------|-----------|--------------------|----|----|
| 1             | SETOVF | TC<br>TCF | OVERFLOW<br>DANZIG | 1  | 1  |
| 2             |        |           |                    | 2  | 2  |
| 3             |        |           |                    | 3  | 3  |
| 4             |        |           |                    | 4  | 4  |
| 5             |        |           |                    | 5  | 5  |
| 6             |        |           |                    | 6  | 6  |
| 7             |        |           |                    | 7  | 7  |
| 8             |        |           |                    | 8  | 8  |
| 9             |        |           |                    | 9  | 9  |
| 10            |        |           |                    | 10 | 10 |
| 11            |        |           |                    | 11 | 11 |
| 12            |        |           |                    | 12 | 12 |
| 13            |        |           |                    | 13 | 13 |
| 14            |        |           |                    | 14 | 14 |
| 15            |        |           |                    | 15 | 15 |
| 16            |        |           |                    | 16 | 16 |
| 17            |        |           |                    | 17 | 17 |
| 18            |        |           |                    | 18 | 18 |
| 19            |        |           |                    | 19 | 19 |
| 20            |        |           |                    | 20 | 20 |
| 21            |        |           |                    | 21 | 21 |
| 22            |        |           |                    | 22 | 22 |
| 23            |        |           |                    | 23 | 23 |
| 24            |        |           |                    | 24 | 24 |
| 25            |        |           |                    | 25 | 25 |
| 26            |        |           |                    | 26 | 26 |
| 27            |        |           |                    | 27 | 27 |
| 28            |        |           |                    | 28 | 28 |
| 29            |        |           |                    | 29 | 29 |
| 30            |        |           |                    | 30 | 30 |
| 31            |        |           |                    | 31 | 31 |
| 32            |        |           |                    | 32 | 32 |
| 33            |        |           |                    | 33 | 33 |
| 34            |        |           |                    | 34 | 34 |
| 35            |        |           |                    | 35 | 35 |
| 36            |        |           |                    | 36 | 36 |
| 37            |        |           |                    | 37 | 37 |
| 38            |        |           |                    | 38 | 38 |
| 39            |        |           |                    | 39 | 39 |
| 40            |        |           |                    | 40 | 40 |
| 41            |        |           |                    | 41 | 41 |
| 42            |        |           |                    | 42 | 42 |
| 43            |        |           |                    | 43 | 43 |
| 44            |        |           |                    | 44 | 44 |
| 45            |        |           |                    | 45 | 45 |
| 46            |        |           |                    | 46 | 46 |
| 47            |        |           |                    | 47 | 47 |
| 48            |        |           |                    | 48 | 48 |
| 49            |        |           |                    | 49 | 49 |
| 50            |        |           |                    | 50 | 50 |
| 51            |        |           |                    | 51 | 51 |
| 52            |        |           |                    | 52 | 52 |
| 53            |        |           |                    | 53 | 53 |
| 54            |        |           |                    | 54 | 54 |
| 55            |        |           |                    | 55 | 55 |
| 56            |        |           |                    | 56 | 56 |
| 57            |        |           |                    | 57 | 57 |
| 58            |        |           |                    | 58 | 58 |
| 59            |        |           |                    | 59 | 59 |
| 60            |        |           |                    | 60 | 60 |

1412THE

# INTERPRETER

|    |          |        |                               |    |
|----|----------|--------|-------------------------------|----|
| 1  |          |        |                               | 1  |
| 2  | DSU      | EXTEND |                               | 2  |
| 3  |          | INDEX  | ADDRWD                        | 3  |
| 4  |          | DCS    | 0                             | 4  |
| 5  |          | TCF    | ENDVXV                        | 5  |
| 6  |          |        |                               | 6  |
| 7  | OVERFLWZ | TS     | L                             | 7  |
| 8  |          | CAF    | FIVE                          | 8  |
| 9  |          | TCF    | +3                            | 9  |
| 10 |          |        | # ENTRY FOR THIRD COMPONENT.  | 10 |
| 11 | OVERFLWY | TS     | L                             | 11 |
| 12 |          | CAF    | THREE                         | 12 |
| 13 |          | XCH    | L                             | 13 |
| 14 |          |        | # ENTRY FOR SECOND COMPONENT. | 14 |
| 15 | OVERFLOW | INDEX  | A                             | 15 |
| 16 |          | CS     | LIMITS                        | 16 |
| 17 |          | TS     | BUF                           | 17 |
| 18 |          | EXTEND |                               | 18 |
| 19 |          | AUG    | A                             | 19 |
| 20 |          | INDEX  | L                             | 20 |
| 21 |          | ADS    | MPAC +1                       | 21 |
| 22 |          | TS     | 7                             | 22 |
| 23 |          | CAF    | ZERO                          | 23 |
| 24 |          | AD     | BUF                           | 24 |
| 25 |          | INDEX  | L                             | 25 |
| 26 |          | ADS    | MPAC                          | 26 |
| 27 |          | TS     | 7                             | 27 |
| 28 |          | TC     | Q                             | 28 |
| 29 |          | TCF    | SETOVF2                       | 29 |
| 30 |          |        | # NO OVERFLOW EXIT.           | 30 |
| 31 |          |        | # SET OVFind AND EXIT.        | 31 |
| 32 | BVSU     | EXTEND |                               | 32 |
| 33 |          | INDEX  | ADDRWD                        | 33 |
| 34 |          | DCA    | 2                             | 34 |
| 35 |          | DXCH   | MPAC +3                       | 35 |
| 36 |          | EXTEND |                               | 36 |
| 37 |          | DCOM   |                               | 37 |
| 38 |          | DAS    | MPAC +3                       | 38 |
| 39 |          | EXTEND |                               | 39 |
| 40 |          | BZF    | +2                            | 40 |
| 41 |          | TC     | OVERFLWY                      | 41 |
| 42 |          | EXTEND |                               | 42 |
| 43 |          | INDEX  | ADDRWD                        | 43 |
| 44 |          | DCA    | 4                             | 44 |
| 45 |          | DXCH   | MPAC +5                       | 45 |
| 46 |          | EXTEND |                               | 46 |
| 47 |          | DCOM   |                               | 47 |
| 48 |          | DAS    | MPAC +5                       | 48 |
| 49 |          | EXTEND |                               | 49 |
| 50 |          | BZF    | +2                            | 50 |
| 51 |          | TC     | OVERFLWZ                      | 51 |
| 52 |          |        |                               | 52 |
| 53 |          |        |                               | 53 |
| 54 |          |        |                               | 54 |
| 55 |          |        |                               | 55 |
| 56 |          |        |                               | 56 |
| 57 |          |        |                               | 57 |
| 58 |          |        |                               | 58 |
| 59 |          |        |                               | 59 |
| 60 |          |        |                               | 60 |



|    |      |        |        |    |
|----|------|--------|--------|----|
| 1  |      |        |        | 1  |
| 2  | BDSU | EXTEND |        | 2  |
| 3  |      | INDEX  | ADDRWD | 3  |
| 4  |      | DCA    | 0      | 4  |
| 5  |      | DXCH   | MPAC   | 5  |
| 6  |      | EXTEND |        | 6  |
| 7  |      | DCOM   |        | 7  |
| 8  |      | TCF    | ENDVXV | 8  |
| 9  |      |        |        | 9  |
| 10 |      |        |        | 10 |
| 11 |      |        |        | 11 |
| 12 |      |        |        | 12 |
| 13 |      |        |        | 13 |
| 14 |      |        |        | 14 |
| 15 |      |        |        | 15 |
| 16 |      |        |        | 16 |
| 17 |      |        |        | 17 |
| 18 |      |        |        | 18 |
| 19 |      |        |        | 19 |
| 20 |      |        |        | 20 |
| 21 |      |        |        | 21 |
| 22 |      |        |        | 22 |
| 23 |      |        |        | 23 |
| 24 |      |        |        | 24 |
| 25 |      |        |        | 25 |
| 26 |      |        |        | 26 |
| 27 |      |        |        | 27 |
| 28 |      |        |        | 28 |
| 29 |      |        |        | 29 |
| 30 |      |        |        | 30 |
| 31 |      |        |        | 31 |
| 32 |      |        |        | 32 |
| 33 |      |        |        | 33 |
| 34 |      |        |        | 34 |
| 35 |      |        |        | 35 |
| 36 |      |        |        | 36 |
| 37 |      |        |        | 37 |
| 38 |      |        |        | 38 |
| 39 |      |        |        | 39 |
| 40 |      |        |        | 40 |
| 41 |      |        |        | 41 |
| 42 |      |        |        | 42 |
| 43 |      |        |        | 43 |
| 44 |      |        |        | 44 |
| 45 |      |        |        | 45 |
| 46 |      |        |        | 46 |
| 47 |      |        |        | 47 |
| 48 |      |        |        | 48 |
| 49 |      |        |        | 49 |
| 50 |      |        |        | 50 |
| 51 |      |        |        | 51 |
| 52 |      |        |        | 52 |
| 53 |      |        |        | 53 |
| 54 |      |        |        | 54 |
| 55 |      |        |        | 55 |
| 56 |      |        |        | 56 |
| 57 |      |        |        | 57 |
| 58 |      |        |        | 58 |
| 59 |      |        |        | 59 |
| 60 |      |        |        | 60 |



|    |                                 |        |        |                              |    |
|----|---------------------------------|--------|--------|------------------------------|----|
| 1  | # TRIPLE PRECISION ADD ROUTINE. |        |        |                              | 1  |
| 2  |                                 |        |        |                              | 2  |
| 3  |                                 |        |        |                              | 3  |
| 4  | TAD                             | EXTEND | ADDRWD |                              | 4  |
| 5  |                                 | INDEX  | 1      |                              | 5  |
| 6  |                                 | DCA    |        | # ADD MINOR PARTS FIRST.     | 6  |
| 7  |                                 | DAS    | MPAC   | +1                           | 7  |
| 8  |                                 | INDEX  | ADDRWD |                              | 8  |
| 9  |                                 | AD     | 0      |                              | 9  |
| 10 |                                 | AD     | MPAC   |                              | 10 |
| 11 |                                 | TS     | MPAC   |                              | 11 |
| 12 |                                 | TCF    | DANZIG |                              | 12 |
| 13 |                                 |        |        |                              | 13 |
| 14 |                                 | TCF    | SETOVF | # SET OVFind IF SUCH OCCURS. | 14 |
| 15 |                                 |        |        |                              | 15 |
| 16 |                                 |        |        |                              | 16 |
| 17 |                                 |        |        |                              | 17 |
| 18 |                                 |        |        |                              | 18 |
| 19 |                                 |        |        |                              | 19 |
| 20 |                                 |        |        |                              | 20 |
| 21 |                                 |        |        |                              | 21 |
| 22 |                                 |        |        |                              | 22 |
| 23 |                                 |        |        |                              | 23 |
| 24 |                                 |        |        |                              | 24 |
| 25 |                                 |        |        |                              | 25 |
| 26 |                                 |        |        |                              | 26 |
| 27 |                                 |        |        |                              | 27 |
| 28 |                                 |        |        |                              | 28 |
| 29 |                                 |        |        |                              | 29 |
| 30 |                                 |        |        |                              | 30 |
| 31 |                                 |        |        |                              | 31 |
| 32 |                                 |        |        |                              | 32 |
| 33 |                                 |        |        |                              | 33 |
| 34 |                                 |        |        |                              | 34 |
| 35 |                                 |        |        |                              | 35 |
| 36 |                                 |        |        |                              | 36 |
| 37 |                                 |        |        |                              | 37 |
| 38 |                                 |        |        |                              | 38 |
| 39 |                                 |        |        |                              | 39 |
| 40 |                                 |        |        |                              | 40 |
| 41 |                                 |        |        |                              | 41 |
| 42 |                                 |        |        |                              | 42 |
| 43 |                                 |        |        |                              | 43 |
| 44 |                                 |        |        |                              | 44 |
| 45 |                                 |        |        |                              | 45 |
| 46 |                                 |        |        |                              | 46 |
| 47 |                                 |        |        |                              | 47 |
| 48 |                                 |        |        |                              | 48 |
| 49 |                                 |        |        |                              | 49 |
| 50 |                                 |        |        |                              | 50 |
| 51 |                                 |        |        |                              | 51 |
| 52 |                                 |        |        |                              | 52 |
| 53 |                                 |        |        |                              | 53 |
| 54 |                                 |        |        |                              | 54 |
| 55 |                                 |        |        |                              | 55 |
| 56 |                                 |        |        |                              | 56 |
| 57 |                                 |        |        |                              | 57 |
| 58 |                                 |        |        |                              | 58 |
| 59 |                                 |        |        |                              | 59 |
| 60 |                                 |        |        |                              | 60 |

# ARITHMETIC SUBROUTINES REQUIRED IN FIXED-FIXED.

# 1. DMPSUB DOUBLE PRECISION MULTIPLY, MULTIPLY THE CONTENTS OF MPAC,+1 BY THE DP WORD WHOSE ADDRESS  
# IS IN ADDRWD AND LEAVE A TRIPLE-PRECISION RESULT IN MPAC.  
# 2. ROUNDSUB ROUND THE TRIPLE PRECISION CONTENTS OF MPAC TO DOUBLE PRECISION.  
# 3. DOTSUB TAKE THE DOT PRODUCT OF THE VECTOR IN MPAC AND THE VECTOR WHOSE ADDRESS IS IN ADDRWD  
# AND LEAVE THE TRIPLE PRECISION RESULT IN MPAC.  
# 4. POLY USING THE CONTENTS OF MPAC AS A DP ARGUMENT, EVALUATE THE POLYNOMIAL WHOSE DEGREE AND  
# COEFFICIENTS IMMEDIATELY FOLLOW THE TC POLY INSTRUCTION (SEE ROUTINE FOR DETAILS).

DMP INDEX Q # BASIC SUBROUTINE FOR USE BY PINBALL, ETC  
CAF 0 # ADRES OF ARGUMENT FOLLOWS TC DMP .

-1 INCR Q  
TS ADDRWD # (PROLOGUE FOR SETTING ADDRWD.)

DMPSUB INDEX ADDRWD # GET MINOR PART OF OPERAND AT C(ADDRWD).  
CA 1  
TS MPAC +2 # THIS WORKS FOR SQUARING MPAC AS WELL.  
CAF ZERO # SET MPAC +1 TO ZERO SO WE CAN ACCUMULATE  
XCH MPAC +1 # THE PARTIAL PRODUCTS WITH DAS  
TS MPTEMP # INSTRUCTIONS.

EXTEND  
MP MPAC +2 # MINOR OF MPAC X MINOR OF C(ADDRWD).

XCH MPAC +2 # DISCARD MINOR PART OF ABOVE RESULT AND  
EXTEND # FORM MAJOR OF MPAC X MINOR OF C(ADDRWD).  
MP MPAC

DAS MPAC +1 # GUARANTEED NO OVERFLOW.

INDEX ADDRWD # GET MAJOR PART OF ARGUMENT AT C(ADDRWD).

DMPSUB2 CA 0  
XCH MPTEMP # SAVE AND BRING OUT MINOR OF MPAC.  
EXTEND

MP MPTEMP # MAJOR OF C(ADDRWD) X MINOR OF MPAC.  
DAS MPAC +1 # ACCUMULATE, SETTING A TO NET OVERFLOW.

XCH MPAC # SETTING MPAC TO 0 OR +-1.

EXTEND # MAJOR OF MPAC X MAJOR OF C(ADDRWD).  
MP MPTEMP

DAS MPAC # GUARANTEED NO OVERFLOW.  
TC Q # 49 MCT = .573 MS. INCLUDING RETURN.





|    |                                                                                 |     |        |                                                  |    |
|----|---------------------------------------------------------------------------------|-----|--------|--------------------------------------------------|----|
| 1  | # ROUND MPAC TO DOUBLE PRECISION, SETTING OVFPND ON THE RARE EVENT OF OVERFLOW. |     |        |                                                  | 1  |
| 2  |                                                                                 |     |        |                                                  | 2  |
| 3  |                                                                                 |     |        |                                                  | 3  |
| 4  | ROUNDSUB                                                                        | CAF | ZERO   | # SET MPAC +2 = 0 FOR SCALARS AND CHANGE         | 4  |
| 5  | +1                                                                              | TS  | MODE   | # MODE TO DP.                                    | 5  |
| 6  |                                                                                 |     |        |                                                  | 6  |
| 7  | VROUND                                                                          | XCH | MPAC   | +2 # BUT WE NEEDN'T TAKE THE TIME FOR VECTORS.   | 7  |
| 8  |                                                                                 |     | DOUBLE |                                                  | 8  |
| 9  |                                                                                 |     | TS     | L                                                | 9  |
| 10 |                                                                                 |     | TC     | Q                                                | 10 |
| 11 |                                                                                 |     |        |                                                  | 11 |
| 12 |                                                                                 |     | AD     | MPAC +1 # ADD ROUDING BIT IF MPAC +2 WAS GREATER | 12 |
| 13 |                                                                                 |     | TS     | MPAC +1 # THAN .5 IN MAGNITUDE.                  | 13 |
| 14 |                                                                                 |     | TC     | Q                                                | 14 |
| 15 |                                                                                 |     |        |                                                  | 15 |
| 16 |                                                                                 |     | AD     | MPAC # PROPAGATE INTERFLOW.                      | 16 |
| 17 |                                                                                 |     | TS     | MPAC                                             | 17 |
| 18 |                                                                                 |     | TC     | Q                                                | 18 |
| 19 |                                                                                 |     |        |                                                  | 19 |
| 20 | SETOVF2                                                                         | TS  | OVFPND | # (RARE).                                        | 20 |
| 21 |                                                                                 |     | TC     | Q                                                | 21 |
| 22 |                                                                                 |     |        |                                                  | 22 |
| 23 |                                                                                 |     |        |                                                  | 23 |
| 24 |                                                                                 |     |        |                                                  | 24 |
| 25 |                                                                                 |     |        |                                                  | 25 |
| 26 |                                                                                 |     |        |                                                  | 26 |
| 27 |                                                                                 |     |        |                                                  | 27 |
| 28 |                                                                                 |     |        |                                                  | 28 |
| 29 |                                                                                 |     |        |                                                  | 29 |
| 30 |                                                                                 |     |        |                                                  | 30 |
| 31 |                                                                                 |     |        |                                                  | 31 |
| 32 |                                                                                 |     |        |                                                  | 32 |
| 33 |                                                                                 |     |        |                                                  | 33 |
| 34 |                                                                                 |     |        |                                                  | 34 |
| 35 |                                                                                 |     |        |                                                  | 35 |
| 36 |                                                                                 |     |        |                                                  | 36 |
| 37 |                                                                                 |     |        |                                                  | 37 |
| 38 |                                                                                 |     |        |                                                  | 38 |
| 39 |                                                                                 |     |        |                                                  | 39 |
| 40 |                                                                                 |     |        |                                                  | 40 |
| 41 |                                                                                 |     |        |                                                  | 41 |
| 42 |                                                                                 |     |        |                                                  | 42 |
| 43 |                                                                                 |     |        |                                                  | 43 |
| 44 |                                                                                 |     |        |                                                  | 44 |
| 45 |                                                                                 |     |        |                                                  | 45 |
| 46 |                                                                                 |     |        |                                                  | 46 |
| 47 |                                                                                 |     |        |                                                  | 47 |
| 48 |                                                                                 |     |        |                                                  | 48 |
| 49 |                                                                                 |     |        |                                                  | 49 |
| 50 |                                                                                 |     |        |                                                  | 50 |
| 51 |                                                                                 |     |        |                                                  | 51 |
| 52 |                                                                                 |     |        |                                                  | 52 |
| 53 |                                                                                 |     |        |                                                  | 53 |
| 54 |                                                                                 |     |        |                                                  | 54 |
| 55 |                                                                                 |     |        |                                                  | 55 |
| 56 |                                                                                 |     |        |                                                  | 56 |
| 57 |                                                                                 |     |        |                                                  | 57 |
| 58 |                                                                                 |     |        |                                                  | 58 |
| 59 |                                                                                 |     |        |                                                  | 59 |
| 60 |                                                                                 |     |        |                                                  | 60 |

# THE DOT PRODUCT SUBROUTINE USUALLY FORMS THE DOT PRODUCT OF THE VECTOR IN MPAC WITH A STANDARD SIX  
# REGISTER VECTOR WHOSE ADDRESS IS IN ADDRWD. IN THIS CASE C(DOTINC) ARE SET TO 2. VXM, HOWEVER, SETS C(DOTINC) TO  
# 6 SO THAT DOTSUB DOTS MPAC WITH A COLUMN VECTOR OF THE MATRIX IN QUESTION IN THIS CASE.

PREDOT CAF TWO # PROLOGUE TO SET DOTINC TO 2.

TS DOTINC

DOTSUB EXTEND

QXCH DOTRET # SAVE RETURN

TC DMPSUB # DOT X COMPONENTS.

DXCH MPAC +3 # POSITION Y COMPONENT OF MPAC FOR

DXCH MPAC # MULTIPLICATION WHILE SAVING RESULT IN

DXCH BUF # THREE WORD BUFFER, BUF.

CA MPAC +2

TS BUF +2

CA DOTINC # ADVANCE ADDRWD TO Y COMPONENT OF

ADS ADDRWD # OTHER ARGUMENT.

TC DMPSUB

DXCH MPAC +1 # ACCUMULATE PARTIAL PRODUCTS.

DAS BUF +1

AD MPAC

AD BUF

TS BUF

TCF +2

TS OVFLND # IF OVERFLOW OCCURS.

DXCH MPAC +5 # MULTIPLY Z COMPONENTS.

DXCH MPAC

CA DOTINC

ADS ADDRWD

TC DMPSUB

ENDDOT DXCH BUF +1 # LEAVE FINAL ACCUMULATION IN MPAC.

DAS MPAC +1

AD MPAC

AD BUF

TS MPAC

TC DOTRET

TC OVERFLOW # ON OVERFLOW HERE.

TC DOTRET

# DOUBLE PRECISION POLYNOMIAL EVALUATOR

#  
$$A_N X^N + A_{N-1} X^{N-1} + \dots + A_1 X + A_0$$
# THIS ROUTINE EVALUATES  $A_N X^N + A_{N-1} X^{N-1} + \dots + A_1 X + A_0$  LEAVING THE DP RESULT IN MPAC ON EXIT.

# THE ROUTINE HAS TWO ENTRIES

# 1 ENTRY THRU POWRSERS. THE COEFFICIENTS MAY BE EITHER IN FIXED OR ERASABLE E. THE CALL IS BY TC POWRSERS, AND THE RETURN IS TO LOC(TC POWRSERS)+1. THE ENTERING DATA MUST BE AS FOLLOWS:

|        |     |       |                                       |
|--------|-----|-------|---------------------------------------|
| A      | SP  | LOC-3 | # ADDRESS FOR REFERENCING COEF TABLE  |
| L      | SP  | N-1   | # N IS THE DEGREE OF THE POWER SERIES |
| MPAC   | DP  | X     | # ARGUMENT                            |
| LOC-2N | DP  | A(0)  |                                       |
| LOC    | ... | A(N)  |                                       |

# 2. ENTRY THRU POLY. THE CALL TO POLY AND THE ENTERING DATA MUST BE AS FOLLOWS

|          |     |      |                         |
|----------|-----|------|-------------------------|
| MPAC     | DP  | X    | # ARGUMENT              |
| LOC      | TC  | POLY |                         |
| LOC+1    | SP  | N-1  |                         |
| LOC+2    | DP  | A(0) |                         |
| LOC+2N+2 | ... | A(N) | # RETURN IS TO LOC+2N+4 |

|          |        |         |                        |
|----------|--------|---------|------------------------|
| POWRSERS | EXTEND |         |                        |
|          | QXCH   | POLYRET | # RETURN ADDRESS       |
|          | TS     | POLISH  | # POWER SERIES ADDRESS |
|          | LXCH   | POLYCNT | # N-1 TO COUNTER       |
|          | TCF    | POLYCOM | # SKIP SET UP BY POLY  |

|      |       |         |                  |
|------|-------|---------|------------------|
| POLY | INDEX | Q       |                  |
|      | CAF   | 0       |                  |
|      | TS    | POLYCNT | # N-1 TO COUNTER |

|  |        |         |                        |
|--|--------|---------|------------------------|
|  | DOUBLE |         |                        |
|  | AD     | Q       |                        |
|  | TS     | POLISH  | # L(A(N))-3 TO POLISH  |
|  | AD     | FIVE    |                        |
|  | TS     | POLYRET | # STORE RETURN ADDRESS |

|         |     |        |                                          |
|---------|-----|--------|------------------------------------------|
| POLYCOM | CAF | LVBUF  | # INCOMING X WILL BE MOVED TO VBUF, SO   |
|         | TS  | ADDRWD | # SET ADDRWD SO DMPSUB WILL MPY BY VBUF. |

|  |        |        |  |
|--|--------|--------|--|
|  | EXTEND |        |  |
|  | INDEX  | POLISH |  |
|  | DCA    | 3      |  |



|    |          |        |          |                                             |    |
|----|----------|--------|----------|---------------------------------------------|----|
| 1  |          |        |          |                                             | 1  |
| 2  |          | DXCH   | MPAC     | # LOAD A(N) INTO MPAC                       | 2  |
| 3  |          | DXCH   | VBUF     | # SAVING X IN VBUF                          | 3  |
| 4  |          | TCF    | POLY2    |                                             | 4  |
| 5  |          |        |          |                                             | 5  |
| 6  | POLYLOOP | TS     | POLYCNT  | # SAVE DECREMENTD LOOP COUNTER              | 6  |
| 7  |          | CS     | TWO      |                                             | 7  |
| 8  |          | ADS    | POLISH   | # REGRESS COEFFICIENT POINTER               | 8  |
| 9  |          |        |          |                                             | 9  |
| 10 | POLY2    | TC     | DMPSUB   | # MULTIPLY BY X                             | 10 |
| 11 |          | EXTEND |          |                                             | 11 |
| 12 |          | INDEX  | POLISH   |                                             | 12 |
| 13 |          | DCA    | 1        | # ADD IN NEXT COEFFICIENT                   | 13 |
| 14 |          | DAS    | MPAC     | # USER'S RESPONSIBILITY TO ASSURE NO OVFLOW | 14 |
| 15 |          |        |          |                                             | 15 |
| 16 |          | CCS    | POLYCNT  |                                             | 16 |
| 17 |          | TCF    | POLYLOOP |                                             | 17 |
| 18 |          | TC     | POLYRET  | # RETURN CALLER                             | 18 |
| 19 |          |        |          |                                             | 19 |
| 20 |          |        |          |                                             | 20 |
| 21 |          |        |          |                                             | 21 |
| 22 |          |        |          |                                             | 22 |
| 23 |          |        |          |                                             | 23 |
| 24 |          |        |          |                                             | 24 |
| 25 |          |        |          |                                             | 25 |
| 26 |          |        |          |                                             | 26 |
| 27 |          |        |          |                                             | 27 |
| 28 |          |        |          |                                             | 28 |
| 29 |          |        |          |                                             | 29 |
| 30 |          |        |          |                                             | 30 |
| 31 |          |        |          |                                             | 31 |
| 32 |          |        |          |                                             | 32 |
| 33 |          |        |          |                                             | 33 |
| 34 |          |        |          |                                             | 34 |
| 35 |          |        |          |                                             | 35 |
| 36 |          |        |          |                                             | 36 |
| 37 |          |        |          |                                             | 37 |
| 38 |          |        |          |                                             | 38 |
| 39 |          |        |          |                                             | 39 |
| 40 |          |        |          |                                             | 40 |
| 41 |          |        |          |                                             | 41 |
| 42 |          |        |          |                                             | 42 |
| 43 |          |        |          |                                             | 43 |
| 44 |          |        |          |                                             | 44 |
| 45 |          |        |          |                                             | 45 |
| 46 |          |        |          |                                             | 46 |
| 47 |          |        |          |                                             | 47 |
| 48 |          |        |          |                                             | 48 |
| 49 |          |        |          |                                             | 49 |
| 50 |          |        |          |                                             | 50 |
| 51 |          |        |          |                                             | 51 |
| 52 |          |        |          |                                             | 52 |
| 53 |          |        |          |                                             | 53 |
| 54 |          |        |          |                                             | 54 |
| 55 |          |        |          |                                             | 55 |
| 56 |          |        |          |                                             | 56 |
| 57 |          |        |          |                                             | 57 |
| 58 |          |        |          |                                             | 58 |
| 59 |          |        |          |                                             | 59 |
| 60 |          |        |          |                                             | 60 |

# MISCELLANEOUS MULTI-PRECISION ROUTINES REQUIRED IN FIXED-FIXED BUT NOT USED BY THE INTERPRETER.

DPAGREE      CAF      ZERO      # DOUBLE PRECISION ENTRY --  
             TS      MPAC      +2      # ZERO LOW-ORDER WORD

TPAGREE      LXCH      Q      # FORCE SIGN AGREEMENT AMONG THE TRIPLE  
             TC      BRANCH      # PRECISION CONTENTS OF MPAC. RETURNING  
             TCF      ARG+      # WITH SIGNUM OF THE INPUT IN A.  
             TCF      ARGZERO

             CS      POSMAX      # IF NEGATIVE.  
             TCF      +2

ARG+      CAF      POSMAX  
             TS      Q

             EXTEND  
             AUG      A      # FORMS +-1.0.

             AD      MPAC      +2  
             TS      MPAC      +2  
             CAF      ZERO

             AD      Q  
             AD      MPAC      +1  
             TS      MPAC      +1

             CAF      ZERO  
             AD      Q      # Q STILL HAS POSMAX OR NEGMAX IN IT.  
             AD      MPAC

ARGZERO2      TS      MPAC      # ALWAYS SKIPPING UNLESS ARGZERO.  
             TS      MPAC      +1  
             TC      L      # RETURN VIA L.

ARGZERO      TS      MPAC      +2      # SET ALL THREE MPAC REGISTERS TO ZERO.  
             TCF      ARGZERO2

# SHORTMP MULTIPLIES THE TP CONTENTS OF MPAC BY THE SINGLE PRECISION NUMBER ARRIVING IN A.

SHORTMP      TS      MPTMP  
             EXTEND

             MP      MPAC      +2

SHORTMP2      TS      MPAC      +2  
             CAF      ZERO      # SO SUBSEQUENT DAS WILL WORK.  
             XCH      MPAC      +1

             TCF      DMPSUB2

# INTERPRETER

```
DMPNSUB MULTIPLIES THE DP FRACTION ARRIVING IN MPAC BY THE SP
INTEGER ARRIVING IN A. THE DP PRODUCT DEPARTS BOTH IN MPAC AND IN
A AND L. NOTE THAT DMPNSUB NORMALLY INCREASES THE MAGNITUDE OF THE
CONTENTS OF MPAC. THE CUSTOMER MUST INSURE THAT B(A) X B(MPAC,MPAC+1)
AND B(A) X B(MPAC) ARE LESS THAN 1 IN MAGNITUDE, WHERE B, AS IS OBVIOUS,
INDICATES THE ARRIVING CONTENTS.

DMPNSUB TS DMPNTEMP
 EXTEND
 MP MPAC +1
 DXCH MPAC # LOW PRODUCT TO MPAC, HIGH FACTOR TO A
 EXTEND
 MP DMPNTEMP
 CA L
 ADS MPAC # COMPLETING THE PRODUCT IN MPAC
 EXTEND
 DCA MPAC # BRINGING THE PRODUCT INTO A AND L
 TC Q
```

1412THE  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
1

# INTERPRETER

|    |                                                                     |     |        |                                            |
|----|---------------------------------------------------------------------|-----|--------|--------------------------------------------|
| 1  | # MISCELLANEOUS VECTOR OPERATIONS. INCLUDED HERE ARE THE FOLLOWING. |     |        |                                            |
| 2  | #                                                                   | 1.  | DOT    | DP VECTOR DOT PRODUCT.                     |
| 3  | #                                                                   | 2.  | VXV    | DP VECTOR CROSS PRODUCT.                   |
| 4  | #                                                                   | 3.  | VXSC   | DP VECTOR TIMES SCALAR.                    |
| 5  | #                                                                   | 4.  | V/SC   | DP VECTOR DIVIDED BY SCALAR.               |
| 6  | #                                                                   | 5.  | VPROJ  | DP VECTOR PROJECTION. ( (MPAC.X)MPAC ).    |
| 7  | #                                                                   | 6.  | VXM    | DP VECTOR POST-MULTIPLIED BY MATRIX.       |
| 8  | #                                                                   | 7.  | MXV    | DP VECTOR PRE-MULTIPLIED BY MATRIX.        |
| 9  |                                                                     |     |        |                                            |
| 10 | DOT                                                                 | TC  | PREDOT | # DO THE DOT PRODUCT AND EXIT, CHANGING    |
| 11 | Dmode                                                               | CAF | ZERO   | # THE MODE TO DP SCALAR.                   |
| 12 |                                                                     |     |        |                                            |
| 13 |                                                                     |     |        |                                            |
| 14 |                                                                     |     |        |                                            |
| 15 | MXV                                                                 | CAF | TWO    | # SET UP MATINC AND DOTINC FOR ROW         |
| 16 |                                                                     |     |        |                                            |
| 17 |                                                                     |     |        |                                            |
| 18 |                                                                     |     |        |                                            |
| 19 | VXM                                                                 | CS  | TEN    | # SET MATINC AND DOTINC TO REFER TO MATRIX |
| 20 |                                                                     |     |        |                                            |
| 21 |                                                                     |     |        |                                            |
| 22 |                                                                     |     |        |                                            |
| 23 |                                                                     |     |        |                                            |
| 24 |                                                                     |     |        |                                            |
| 25 |                                                                     |     |        |                                            |
| 26 |                                                                     |     |        |                                            |
| 27 |                                                                     |     |        |                                            |
| 28 |                                                                     |     |        |                                            |
| 29 |                                                                     |     |        |                                            |
| 30 |                                                                     |     |        |                                            |
| 31 |                                                                     |     |        |                                            |
| 32 |                                                                     |     |        |                                            |
| 33 |                                                                     |     |        |                                            |
| 34 |                                                                     |     |        |                                            |
| 35 |                                                                     |     |        |                                            |
| 36 |                                                                     |     |        |                                            |
| 37 |                                                                     |     |        |                                            |
| 38 |                                                                     |     |        |                                            |
| 39 |                                                                     |     |        |                                            |
| 40 |                                                                     |     |        |                                            |
| 41 |                                                                     |     |        |                                            |
| 42 |                                                                     |     |        |                                            |
| 43 |                                                                     |     |        |                                            |
| 44 |                                                                     |     |        |                                            |
| 45 |                                                                     |     |        |                                            |
| 46 |                                                                     |     |        |                                            |
| 47 |                                                                     |     |        |                                            |
| 48 |                                                                     |     |        |                                            |
| 49 |                                                                     |     |        |                                            |
| 50 |                                                                     |     |        |                                            |
| 51 |                                                                     |     |        |                                            |
| 52 |                                                                     |     |        |                                            |
| 53 |                                                                     |     |        |                                            |
| 54 |                                                                     |     |        |                                            |
| 55 |                                                                     |     |        |                                            |
| 56 |                                                                     |     |        |                                            |
| 57 |                                                                     |     |        |                                            |
| 58 |                                                                     |     |        |                                            |
| 59 |                                                                     |     |        |                                            |
| 60 |                                                                     |     |        |                                            |

# INTERPRETER

# COMMON PORTION OF MXV AND VXM.

VXM/MXV TS DOTINC  
# ITRACE (2) REFERS TO "VXM/MXV".  
TC MPACVBUF # SAVE VECTOR IN MPAC FOR FURTHER USE.

TC DOTSUB # GO DOT TO GET X COMPONENT OF ANSWER.  
EXTEND

DCA VBUF # MOVE MPAC VECTOR BACK INTO MPAC, SAVING  
DXCH MPAC # NEW X COMPONENT IN BUF2.  
DXCH BUF2

EXTEND  
DCA VBUF +2  
DXCH MPAC +3

EXTEND  
DCA VBUF +4  
DXCH MPAC +5

CA MATINC # INITIALIZE ADDRWD FOR NEXT DOT PRODUCT.  
ADS ADDRWD # FORMS HAS ADDRESS OF NEXT COLUMN(ROW).

TC DOTSUB  
DXCH VBUF # MORE GIVEN VECTOR BACK TO MPAC, SAVING Y  
DXCH MPAC # COMPONENT OF ANSWER IN VBUF +2.

DXCH VBUF +2  
DXCH MPAC +3  
DXCH VBUF +4  
DXCH MPAC +5

CA MATINC # FORM ADDRESS OF LAST COLUMN OR ROW.  
ADS ADDRWD

TC DOTSUB  
DXCH BUF2 # ANSWER NOW COMPLETE. PUT COMPONENTS INTO  
DXCH MPAC # PROPER MPAC REGISTERS.

DXCH MPAC +5  
DXCH VBUF +2  
DXCH MPAC +3  
TCF DANZIG

# EXIT.



# INTERPRETER

# VXSC -- VECTOR TIMES SCALAR.

|      |     |       |                                            |
|------|-----|-------|--------------------------------------------|
| VXSC | CCS | MODE  | # TEST PRESENT MODE.                       |
|      | TCF | DVXSC | # SEPARATE ROUTINE WHEN SCALAR IS IN MPAC. |
|      | TCF | DVXSC |                                            |

|       |      |        |                                            |
|-------|------|--------|--------------------------------------------|
| VVXSC | TC   | DMPSUB | # COMPUTE X COMPONENT                      |
|       | TC   | VROUND | # AND ROUND IT.                            |
|       | DXCH | MPAC   | +3                                         |
|       | DXCH | MPAC   | # PUT Y COMPONENT INTO MPAC SAVING MPAC IN |
|       | DXCH | MPAC   | +3                                         |
|       |      |        | # MPAC +3.                                 |

|  |      |        |                                   |
|--|------|--------|-----------------------------------|
|  | TC   | DMPSUB | # DO SAME FOR Y AND Z COMPONENTS. |
|  | TC   | VROUND |                                   |
|  | DXCH | MPAC   | +5                                |
|  | DXCH | MPAC   |                                   |
|  | DXCH | MPAC   | +5                                |

|          |      |        |                                            |
|----------|------|--------|--------------------------------------------|
|          | TC   | DMPSUB |                                            |
|          | TC   | VROUND |                                            |
| VROTATEX | DXCH | MPAC   | # EXIT USED TO RESTORE MPAC AFTER THIS     |
|          | DXCH | MPAC   | +5                                         |
|          | DXCH | MPAC   | # TYPE OF ROTATION. CALLED BY VECTOR SHIFT |
|          | DXCH | MPAC   | +3                                         |
|          |      |        | # RIGHT, V/SC, ETC.                        |
|          | DXCH | MPAC   |                                            |
|          | TCF  | DANZIG |                                            |

# DP VECTOR PROJECTION ROUTINE.

|       |     |        |                                        |
|-------|-----|--------|----------------------------------------|
| VPROJ | TC  | PREDOT | # (MPAC.X)MPAC IS COMPUTED AND LEFT IN |
|       | CS  | FOUR   | # MPAC. DO DOT AND FALL INTO DVXSC.    |
|       | ADS | ADDRWD |                                        |

# VXSC WHEN SCALAR ARRIVES IN MPAC AND VECTOR IS AT X.

|       |        |      |                                    |
|-------|--------|------|------------------------------------|
| DVXSC | EXTEND |      | # SAVE SCALAR IN MPAC +3 AND GET X |
|       | DCA    | MPAC | # COMPONENT OF ANSWER.             |

|      |      |    |
|------|------|----|
| DXCH | MPAC | +3 |
|------|------|----|

|    |        |
|----|--------|
| TC | DMPSUB |
| TC | VROUND |

|        |        |                                       |
|--------|--------|---------------------------------------|
| CAF    | TWO    | # ADVANCE ADDRWD TO Y COMPONENT OF X. |
| ADS    | ADDRWD |                                       |
| EXTEND |        |                                       |

|      |      |    |                                      |
|------|------|----|--------------------------------------|
| DCA  | MPAC | +3 | # PUT SCALAR BACK INTO MPAC AND SAVE |
| DXCH | MPAC |    | # X RESULT IN MPAC +5.               |

|      |      |    |
|------|------|----|
| DXCH | MPAC | +5 |
|------|------|----|

|    |        |
|----|--------|
| TC | DMPSUB |
| TC | VROUND |

|      |        |    |                                          |
|------|--------|----|------------------------------------------|
| CAF  | TWO    |    | # TO Z COMPONENT.                        |
| ADS  | ADDRWD |    | # BRING SCALAR BACK, PUTTING Y RESULT IN |
| DXCH | MPAC   | +3 | # THE PROPER PLACE.                      |

|      |      |    |
|------|------|----|
| DXCH | MPAC | +3 |
|------|------|----|

|    |        |
|----|--------|
| TC | DMPSUB |
| TC | VROUND |

|      |      |    |                                         |
|------|------|----|-----------------------------------------|
| DXCH | MPAC |    | # PUT Z COMPONENT IN PROPER PLACE, ALSO |
| DXCH | MPAC | +5 | # POSITIONING X.                        |

|      |      |
|------|------|
| DXCH | MPAC |
|------|------|

|     |       |                               |
|-----|-------|-------------------------------|
| TCF | VMODE | # MODE HAS CHANGED TO VECTOR. |
|-----|-------|-------------------------------|

# INTERPRETER

# VECTOR CROSS PRODUCT ROUTINE CALCULATES (X M -X M ,X M -X M ,X M -X M ) WHERE M IS THE VECTOR IN  
#  
# MPAC AND X THE VECTOR AT THE GIVEN ADDRESS.

|     |        |          |    |                                            |
|-----|--------|----------|----|--------------------------------------------|
| VXV | EXTEND |          |    |                                            |
|     | DCA    | MPAC     | +5 | # FORM UP M3X1, LEAVING M1 IN VBUF.        |
|     | DXCH   | MPAC     |    |                                            |
|     | DXCH   | VBUF     |    |                                            |
|     | TC     | DMPSUB   |    | # BY X1.                                   |
|     | EXTEND |          |    |                                            |
|     | DCS    | MPAC     | +3 | # CALCULATE -X1M2, SAVING X1M3 IN VBUF +2. |
|     | DXCH   | MPAC     |    |                                            |
|     | DXCH   | VBUF     | +2 |                                            |
|     | TC     | DMPSUB   |    |                                            |
|     | CAF    | TWO      |    | # ADVANCE ADDRWD TO X2.                    |
|     | ADS    | ADDRWD   |    |                                            |
|     | EXTEND |          |    |                                            |
|     | DCS    | MPAC     | +5 | # PREPARE TO GET -X2M3, SAVING -X1M2 IN    |
|     | DXCH   | MPAC     |    | # MPAC +5.                                 |
|     | DXCH   | MPAC     | +5 |                                            |
|     | TC     | DMPSUB   |    |                                            |
|     | EXTEND |          |    |                                            |
|     | DCA    | VBUF     |    | # GET X2M1, SAVING -X2M3 IN VBUF +4.       |
|     | DXCH   | MPAC     |    |                                            |
|     | DXCH   | VBUF     | +4 |                                            |
|     | TC     | DMPSUB   |    |                                            |
|     | CAF    | TWO      |    | # ADVANCE ADDRWD TO X3.                    |
|     | ADS    | ADDRWD   |    |                                            |
|     | EXTEND |          |    |                                            |
|     | DCS    | VBUF     |    | # GET -X3M1, ADDING X2M1 TO MPAC +5 TO     |
|     | DXCH   | MPAC     |    | # COMPLETE THE Z COMPONENT OF THE ANSWER.  |
|     | DAS    | MPAC     | +5 |                                            |
|     | EXTEND |          |    |                                            |
|     | BZF    | +2       |    |                                            |
|     | TC     | OVERFLWZ |    |                                            |
|     | TC     | DMPSUB   |    |                                            |
|     | DXCH   | VBUF     | +2 | # MOVE X1M3 TO MPAC +3 SETTING UP FOR X3M2 |
|     | DXCH   | MPAC     | +3 | # AND ADD -X3M1 TO MPAC +3 TO COMPLETE THE |
|     | DXCH   | MPAC     |    | # Y COMPONENT OF THE RESULT.               |
|     | DAS    | MPAC     | +3 |                                            |
|     | EXTEND |          |    |                                            |
|     | BZF    |          | +2 |                                            |

TC OVERFLWY

```
TC DMPSUB
DXCH VBUF +4 # GO ADD -X2M3 TO X3M2 TO COMPLETE THE X
TCF ENDVXV # COMPONENT (TAIL END OF DAD).
```

# THE MPACVBUF SUBROUTINE SAVES THE VECTOR IN MPAC IN VBUF WITHOUT CLOBBERING MPAC.

MPACVBUF      EXTEND      # CALLED BY MXV, VXM, AND UNIT.

```
DCA MPAC
DXCH VBUF
```

```
EXTEND
DCA MPAC +3
DXCH VBUF +2
```

```
EXTEND
DCA MPAC +5
DXCH VBUF +4
```

TC      Q      # RETURN TO CALLER.

# DOUBLE PRECISION SIGN AGREE ROUTINE. ARRIVE WITH INPUT IN A+L. OUTPUT IS IN A + L.

```
ALSIGNAG CCS A # TEST UPPER PART.
 TCF UPPOS # IT IS POSITIVE
```

```
TC Q # ZERO
TCF UPNEG # NEGATIVE
TC Q # ZERO
```

UPPOS      XCH      L      # SAVE DECREMENTED UPPER PART.

```
AD HALF
AD HALF
TS A
TCF +2
```

```
INCR L # RESTORE UPPER TO ORIGINAL VALUE
XCH L # SWAP A + L BANCK.
TC Q
```

UPNEG      XCH      L      # SAVE COMPLEMENTED + DECREMENTED UPPER PT

```
AD NEGMAX
AD NEGONE
TS A
TCF +2
```

# DON'T INCREMENT IF NO OVERFLOW.

```
INCR L
XCH L
COM # MAKE NEGATIVE AGAIN.
TC Q
```

# INTERPRETER

# INTERPRETIVE INSTRUCTIONS WHOSE EXECUTION CONSISTS OF PRINCIPALLY CALLING SUBROUTINES.

|      |     |        |                    |
|------|-----|--------|--------------------|
| DMP1 | TC  | DMPSUB | # DMP INSTRUCTIONS |
|      | TCF | DANZIG |                    |

|      |     |              |                |
|------|-----|--------------|----------------|
| DMPR | TC  | DMPSUB       |                |
|      | TC  | ROUND SUB +1 | # (C(A) = +0). |
|      | TCF | DANZIG       |                |

|     |        |         |                           |
|-----|--------|---------|---------------------------|
| DDV | EXTEND |         |                           |
|     | INDEX  | ADDRWD  | # MOVE DIVIDEND INTO BUF. |
|     | DCA    | 0       |                           |
|     | TCF    | BDDV +4 |                           |

|      |        |        |                                           |
|------|--------|--------|-------------------------------------------|
| BDDV | EXTEND |        | # MOVE DIVISOR INTO MPAC SAVING MPAC, THE |
|      | INDEX  | ADDRWD | # DIVIDEND, IN BUF.                       |
|      | DCA    | 0      |                                           |

|    |      |          |                              |
|----|------|----------|------------------------------|
| +4 | DXCH | MPAC     |                              |
|    | DXCH | BUF      |                              |
|    | CAF  | ZERO     | # DIVIDE ROUTINES IN BANK 0. |
|    | TS   | FBANK    |                              |
|    | TCF  | DDV/BDDV |                              |

|       |     |          |                                            |
|-------|-----|----------|--------------------------------------------|
| SETPD | CA  | ADDRWD   | # MUST SET TO WORK AREA, OR EBANK TROUBLE. |
|       | TS  | PUSHLOC  |                                            |
|       | TCF | NOIBNKSW | # NO FBANK SWITCH REQUIRED.                |

|      |     |       |                                         |
|------|-----|-------|-----------------------------------------|
| TSLC | CAF | ZERO  | # SHIFTING ROUTINES LOCATED IN BANK 00. |
|      | TS  | FBANK |                                         |
|      | TCF | TSLC2 |                                         |

|        |     |          |                                            |
|--------|-----|----------|--------------------------------------------|
| GSHIFT | CAF | LOW7     | # USED AS MASK AT GENSHIFT. THIS PROCESSES |
|        | TS  | FBANK    | # ANY SHIFT INSTRUCTION (EXCEPT TSLC) WITH |
|        | TCF | GENSHIFT | # AN ADDRESS (ROUTINES IN BANK 0).         |

# THE FOLLOWING IS THE PROLOGUE TO V/SC. IF THE PRESENT MODE IS VECTOR, IT SAVES THE SCALAR AT X IN BUF  
# AND CALLS THE V/SC ROUTINE IN BANK 0. IF THE PRESENT MODE IS SCALAR, IT MOVES THE VECTOR AT X INTO MPAC, SAVING  
# THE SCALAR IN MPAC IN BUF BEFORE CALLING THE V/SC ROUTINE IN BANK 0.

|       |        |        |                                            |
|-------|--------|--------|--------------------------------------------|
| V/SC  | CCS    | MODE   |                                            |
|       | TCF    | DV/SC  | # MOVE VECTOR INTO MPAC.                   |
|       | TCF    | DV/SC  |                                            |
| VV/SC | EXTEND | ADDRWD |                                            |
|       | INDEX  | 0      |                                            |
|       | DCA    |        |                                            |
| V/SC1 | DXCH   | BUF    | # IN BOTH CASES, VECTOR IS NOW IN MPAC AND |
|       | CAF    | ZERO   | # SCALAR IN BUF.                           |
|       | TS     | FBANK  |                                            |
|       | TCF    | V/SC2  |                                            |
| DV/SC | EXTEND | ADDRWD |                                            |
|       | INDEX  | 2      |                                            |
|       | DCA    |        |                                            |
|       | DXCH   | MPAC   | +3                                         |
|       | EXTEND | ADDRWD |                                            |
|       | INDEX  | 4      |                                            |
|       | DCA    |        |                                            |
|       | DXCH   | MPAC   | +5                                         |
|       | CS     | ONE    | # CHANGE MODE TO VECTOR.                   |
|       | TS     | MODE   |                                            |
|       | EXTEND | ADDRWD |                                            |
|       | INDEX  | 0      |                                            |
|       | DCA    |        |                                            |
|       | DXCH   | MPAC   |                                            |
|       | TCF    | V/SC1  | # FINISH PROLOGUE AT COMMON SECTION.       |

# INTERPRETER

# SIGN AND COMPLEMENT INSTRUCTIONS.

SIGN INDEX ADDRWD # CALL COMP INSTRUCTION IF WORD AT X IS  
CCS 0 # NEGATIVE NON-ZERO.  
TCF DANZIG

TCF +2  
TCF COMP # DO THE COMPLEMENT.

CCSL INDEX ADDRWD  
CCS 1  
TCF DANZIG  
TCF DANZIG  
TCF COMP  
TCF DANZIG

COMP EXTEND # COMPLEMENT DP MPAC IN EVERY CASE.  
DCS MPAC  
DXCH MPAC

CCS MODE # EITHER COMPLEMENT MPAC +3 OR THE REST OF  
TCF DCOMP # THE VECTOR ACCUMULATOR.  
TCF DCOMP

EXTEND # VECTOR COMPLEMENT.

DCS MPAC +3  
DXCH MPAC +3  
EXTEND

DCS MPAC +5  
DXCH MPAC +5  
TCF DANZIG

DCOMP CS MPAC +2  
TS MPAC +2  
TCF DANZIG

1412THE

# THE FOLLOWING SHORT SHIFT CODES REQUIRE NO ADDRESS WORD:

|   |    |              |                                     |
|---|----|--------------|-------------------------------------|
| # | 1. | SR1 TO SR4   | SCALAR SHIFT RIGHT.                 |
| # | 2. | SR1R TO SR4R | SCALAR SHIFT RIGHT AND ROUND.       |
| # | 3. | SL1 TO SL4   | SCALAR SHIFT LEFT.                  |
| # | 4. | SL1R TO SL4R | SCALAR SHIFT LEFT AND ROUND.        |
| # | 5. | VSR1 TO VSR8 | VECTOR SHIFT RIGHT (ALWAYS ROUNDS). |
| # | 6. | VSL1 TO VSL8 | VECTOR SHIFT LEFT (NEVER ROUNDS).   |

# THE FOLLOWING CODES REQUIRE AN ADDRESS WHICH MAY BE INDEXED:\*

|   |    |     |                               |
|---|----|-----|-------------------------------|
| # | 1. | SR  | SCALAR SHIFT RIGHT.           |
| # | 2. | SRR | SCALAR SHIFT RIGHT AND ROUND. |
| # | 3. | SL  | SCALAR SHIFT LEFT.            |
| # | 4. | SLR | SCALAR SHIFT LEFT AND ROUND.  |
| # | 5. | VSR | VECTOR SHIFT RIGHT.           |
| # | 6. | VSL | VECTOR SHIFT LEFT.            |

# \* IF THE ADDRESS IS INDEXED, AND THE INDEX MODIFICATION RESULTS IN A NEGATIVE SHIFT COUNT, A SHIFT OF THE  
# ABSOLUTE VALUE OF THE COUNT IS DONE IN THE OPPOSITE DIRECTION.

BANK 00

COUNT\* \$\$/INTER

|        |      |     |                                            |
|--------|------|-----|--------------------------------------------|
| SHORTT | CAF  | SIX | # SCALAR SHORT SHIFTS COME HERE. THE SHIFT |
|        | MASK | CYR | # COUNT-1 IS NOW IN BITS 2-3 OF CYR. THE   |
|        | TS   | SR  | # ROUNDING BIT IS IN BIT1 AT THIS POINT.   |

|     |      |                                       |
|-----|------|---------------------------------------|
| CCS | CYR  | # SEE IF RIGHT OR LEFT SHIFT DESIRED. |
| TCF | TSSL | # SHIFT LEFT.                         |

|       |     |    |                                     |
|-------|-----|----|-------------------------------------|
| SRDDV | DEC | 20 | # MPTEMP SETTING FOR SR BEFORE DDV. |
|-------|-----|----|-------------------------------------|

|      |       |        |                     |
|------|-------|--------|---------------------|
| TSSR | INDEX | SR     | # GET SHIFTING BIT. |
|      | CAF   | BIT14  |                     |
|      | TS    | MPTEMP |                     |

|        |     |          |                                 |
|--------|-----|----------|---------------------------------|
| RIGHTR | CCS | CYR      | # SEE IF A ROUND IS DESIRED.    |
|        | TC  | MPACSRND | # YES -- SHIFT RIGHT AND ROUND. |

|         |        |         |                                      |
|---------|--------|---------|--------------------------------------|
| MPACSHR | TCF    | NEWMODE | # SET MODE TO DP (C(A) = 0).         |
|         | CA     | MPTEMP  | # DO A TRIPLE PRECISION SHIFT RIGHT. |
|         | EXTEND |         |                                      |

|    |    |        |    |                               |
|----|----|--------|----|-------------------------------|
| +3 | MP | MPAC   | +2 |                               |
|    | TS | MPAC   | +2 | # (EXIT FROM SQRT AND ABVAL). |
|    | CA | MPTEMP |    |                               |

|        |      |  |  |                                          |
|--------|------|--|--|------------------------------------------|
| EXTEND |      |  |  |                                          |
| MP     | MPAC |  |  | # SHIFT MAJOR PART INTO A,L AND PLACE IN |



|   |  |
|---|--|
| 1 |  |
| 2 |  |

# INTERPRETER

# ROUTINE FOR SHORT SCALAR SHIFT LEFT (AND MAYBE ROUND).

|            |          |              |                                          |
|------------|----------|--------------|------------------------------------------|
| TSSL<br>+1 | CA<br>TS | SR<br>MPTEMP | # GET SHIFT COUNT FOR SR.                |
| +2         | EXTEND   |              | # ENTRY HERE FROM SL FOR SCALARS.        |
|            | DCA      | MPAC +1      | # SHIFTING LEFT ONE PLACE AT A TIME IS   |
|            | DAS      | MPAC +1      | # FASTER THAN DOING THE WHOLE SHIFT WITH |
|            | AD       | MPAC         | # MULTIPLIES ASSUMING THAT FREQUENCY OF  |
|            | AD       | MPAC         | # SHIFT COUNTS GOES DOWN RAPIDLY AS A    |
|            | TS       | MPAC         | # FUNCTION OF THEIR MAGNITUDE.           |
|            | TCF      | +2           |                                          |
|            | TS       | OVFIND       | # OVERFLOW. (LEAVES OVERFLOW-CORRECTED   |
|            |          |              | # RESULT ANYWAY).                        |
|            | CCS      | MPTEMP       | # LOOP ON DECREMENTED SHIFT COUNT.       |
|            | TCF      | TSSL +1      |                                          |
| ROUND      | CCS      | CYR          | # SEE IF ROUND WANTED.                   |
|            | TC       | ROUND SUB    | # YES -- ROUND AND EXIT.                 |
|            | TCF      | DANZIG       | # SL LEAVES A ZERO IN CYR FOR NO ROUND.  |
|            | TCF      | DANZIG       | # NO -- EXIT IMMEDIATELY                 |

1412THE

# INTERPRETER

# VECTOR SHIFTING ROUTINES.

|        |       |          |                                          |
|--------|-------|----------|------------------------------------------|
| SHORTV | CAF   | LOW3     | # SAVE 3 BIT SHIFT COUNT -- 1 WITHOUT    |
|        | MASK  | CYR      | # EDITING CYR.                           |
|        | TS    | MPTEMP   |                                          |
|        | CCS   | CYR      | # SEE IF LEFT OR RIGHT SHIFT.            |
|        | TCF   | VSSL     | # VECTOR SHIFT LEFT.                     |
| OCT176 | OCT   | 176      | # USED IN PROCESSED SHIFTS WITH - COUNT. |
| VSSR   | INDEX | MPTEMP   | # (ENTRY FROM SR). PICK UP SHIFTING BIT. |
|        | CAF   | BIT14    | # MPTEMP CONTAINS THE SHIFT COUNT - 1.   |
|        | TS    | MPTEMP   |                                          |
|        | TC    | VSHRRND  | # SHIFT X COMPONENT.                     |
|        | DXCH  | MPAC     | # SWAP X AND Y COMPONENTS.               |
|        | DXCH  | MPAC +3  |                                          |
|        | DXCH  | MPAC     |                                          |
|        | TC    | VSHRRND  | # SHIFT Y COMPONENT.                     |
|        | DXCH  | MPAC     | # SWAP Y AND Z COMPONENTS.               |
|        | DXCH  | MPAC +5  |                                          |
|        | DXCH  | MPAC     |                                          |
|        | TC    | VSHRRND  | # SHIFT Z COMPONENT.                     |
|        | TCF   | VROTATEX | # RESTORE COMPONENTS TO PROPER PLACES.   |



```
VECTOR SHIFT LEFT -- DONE ONE PLACE AT A TIME.

-1 TS MPTEMP # SHIFTING LOOP.

VSSL EXTEND
 DCA MPAC
 DAS MPAC
 EXTEND
 BZF +2
 TC OVERFLOW

 EXTEND
 DCA MPAC +3
 DAS MPAC +3
 EXTEND
 BZF +2
 TC OVERFLWY

 EXTEND
 DCA MPAC +5
 DAS MPAC +5
 EXTEND
 BZF +2
 TC OVERFLWZ

 CCS MPTEMP # LOOP ON DECREMENTED SHIFT COUNTER.
 TCF VSSL -1
 TCF DANZIG # EXIT.
```

# INTERPRETER

# TSLC -- TRIPLE SHIFT LEFT AND COUNT. SHIFTS MPAC LEFT UNTIL GREATER THAN .5 IN MAGNITUDE, LEAVING  
# THE COMPLEMENT OF THE NUMBER OF SHIFTS REQUIRED IN X.

TSLC2            TS        MPTEMP        # START BY ZEROING SHIFT COUNT (IN A NOW).  
                 TC        BRANCH        # EXIT WITH NO SHIFTING IF ARGUMENT ZERO.

TCF        +2  
TCF        ENDTSLC        # STORES ZERO SHIFT COUNT IN THIS CASE.

TC        TPAGREE        # MAY CAUSE UPSHIFT OF ONE EXTRA PLACE.

CA        MPAC        # BEGIN NORMALIZATION LOOP.

TCF        TSLCTEST

TSLCLOOP        INCR        MPTEMP        # INCREMENT SHIFT COUNTER.

EXTEND

DCA        MPAC        +1

DAS        MPAC        +1

AD        MPAC

ADS        MPAC

TSLCTEST        DOUBLE        # SEE IF (ANOTHER) SHIFT IS REQUIRED

OVSF

TCF        TSLCLOOP        # YES -- INCREMENT COUNT AND SHIFT AGAIN.

ENDTSLC        CS        MPTEMP  
                 TCF        STORE1        # STORE SHIFT COUNT AND RETURN TO DANZIG.

# INTERPRETER

# THE FOLLOWING ROUTINE PROCESSES THE GENERAL SHIFT INSTRUCTIONS SR, SRR, SL, AND SLR.  
# THE GIVEN ADDRESS IS DECODED AS FOLLOWS:  
#       BITS 1-7       SHIFT COUNT (SUBADDRESS) LESS THAN 125 DECIMAL.  
#       BIT 8         PSEUDO SIGN BIT (DETECTS CHANGE IN SIGN IN INDEXED SHIFTS).  
#       BIT 9         0 FOR LEFT SHIFT, AND 1 FOR RIGHT SHIFT.  
#       BIT 10        1 FOR TERMINAL ROUND ON SCALAR SHIFTS, 0 OTHERWISE  
#       BITS 11-13    0.  
#       BIT 14        1.  
#       BIT 15        0.  
# THE ABOVE ENCODING IS DONE BY THE YUL SYSTEM.

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| GENSHIFT | MASK   | ADDRWD   | # GET SHIFT COUNT, TESTING FOR ZERO.       |
|          | CCS    | A        | # (ARRIVES WITH C(A) = LOW7).              |
|          | TCF    | GENSHFT2 | # IF NON-ZERO, PROCEED WITH DECREMENTED CT |
|          | CAF    | BIT10    | # ZERO SHIFT COUNT. NO SHIFTS NEEDED BUT   |
|          | MASK   | ADDRWD   | # WE MIGHT HAVE TO ROUND MPAC ON SLR AND   |
|          | CCS    | A        | # SRR (SCALAR ONLY).                       |
|          | TC     | ROUNDSUB |                                            |
|          | TCF    | DANZIG   |                                            |
| GENSHFT2 | TS     | MPTEMP   | # DECREMENTED SHIFT COUNT TO MPTEMP.       |
|          | CAF    | BIT8     | # TEST MEANING OF LOW SEVEN BIT COUNT IN   |
|          | EXTEND |          | # MPTEMP NOW.                              |
|          | MP     | ADDRWD   |                                            |
|          | MASK   | LOW2     | # JUMPS ON SHIFT DIRECTION (BIT8) AND      |
|          | INDEX  | A        |                                            |
|          | TCF    | +1       | # ORIGINAL SHIFT DIRECTION (BIT 9)         |
|          | TCF    | RIGHT-   | # NEGATIVE SHIFT COUNT FOR SL OR SLR.      |
|          | TCF    | LEFT     | # SL OR SLR.                               |
|          | TCF    | LEFT-    | # NEGATIVE SHIFT COUNT WITH SR OR SRR.     |

## # GENERAL SHIFT RIGHT

RIGHT CCS MODE # SET IF VECTOR OR SCALAR.

TCF GENSCR  
TCF GENSCRVRIGHT2 CA MPTEMP # SEE IF SHIFT COUNT LESS THAN 14D.  
AD NEG12EXTEND  
BZMF VSSR # IF SO, BRANCH AND SHIFT IMMEDIATELY.AD NEGONE # IF NOT, REDUCE MPTEMP BY A TOTAL OF 14.  
TS MPTEMP # AND DO A SHIFT RIGHT AND ROUND BY 14.  
CAF ZERO # THE ROUND AT THIS STAGE MAY INTRODUCE A  
TS L # ONE BIT ERROR IN A SHIFT RIGHT 15D.XCH MPAC  
XCH MPAC +1TC SETROUND # X COMPONENT NOW SHIFTED, SO MAKE UP THE  
DAS MPAC # ROUNDING QUANTITY (0 IN A AND 0 OR +-1  
# IN L).XCH MPAC +3 # REPEAT THE ABOVE PROCESS FOR Y AND Z.  
XCH MPAC +4

TC SETROUND

DAS MPAC +3 # NO OVERFLOW ON THESE ADDS.

XCH MPAC +5  
XCH MPAC +6TC SETROUND  
DAS MPAC +5CCS MPTEMP # SEE IF DONE, DOING FINAL DECREMENT.  
TS MPTEMPBIASLO TCF VRIGHT2  
DEC .2974 B-1 # SQRT CONSTANT

TCF DANZIG

SETROUND DOUBLE # MAKES UP ROUNDING QUANTITY FROM ARRIVING  
TS MPAC +2 # C(A). L IS ZERO INITIALLY.CAF ZERO  
XCH L

TC Q # RETURN AND DO THE DAS, RESETTNG L TO 0.

# PROCESS SR AND SRR FOR SCALARS.

GENSCR            CA        MPTEMP        # SEE IF THE ORIGINAL SHIFT COUNT WAS LESS  
+1                AD        NEG12        # THAN 14D.

EXTEND

BZMF        DOSSHFT        # DO THE SHIFT IMMEDIATELY IF SO.

+4                AD        NEGONE        # IF NOT, DECREMENT SHIFT COUNT BY 14D AND  
                  TS        MPTEMP        # SHIFT MPAC RIGHT 14 PLACES.

CAF        ZERO  
XCH        MPAC

XCH        MPAC        +1  
TS        MPAC        +2  
CCS        MPTEMP        # SEE IF FINISHED, DO FINAL DECREMENT.

SLOPEHI        TS        MPTEMP  
                  TC        GENSCR    +1  
                  DEC        .5884        # SQRT CONSTANT.

CAF        BIT10        # FINISHED WITH SHIFT. SEE IF ROUND  
MASK       ADDRWD       # WANTED.  
CCS        A

TC        ROUND SUB  
TCF       DANZIG        # DO SO AND/OR EXIT.

DOSSHFT        INDEX    MPTEMP        # PICK UP SHIFTING BIT.

CAF        BIT14  
TS        MPTEMP

CAF        BIT10        # SEE IF TERMINAL ROUND DESIRED.  
MASK       ADDRWD  
CCS        A

TCF        RIGHTR        # YES.  
TCF        MPACSHR       # JUST SHIFT RIGHT.



# INTERPRETER

# PROCESS THE RIGHT- (SL(R) WITH A NEGATIVE COUNT), LEFT-, AND LEFT OPTIONS.

|        |        |         |                                            |
|--------|--------|---------|--------------------------------------------|
| RIGHT- | CS     | MPTEMP  | # GET ABSOLUTE VALUE - 1 OF SHIFT COUNT    |
|        | AD     | OCT176  | # UNDERSTANDING THAT BIT8 (PSEUDO-SIGN)    |
|        | TS     | MPTEMP  | # WAS 1 INITIALLY.                         |
|        | TCF    | RIGHT   | # DO NORMAL SHIFT RIGHT.                   |
| LEFT-  | CS     | OCT176  | # SAME PROLOGUE TO LEFT FOR INDEXED RIGHT  |
|        | AD     | MPTEMP  | # SHIFT WHOSE NET SHIFT COUNT IS NEGATIVE  |
|        | COM    |         |                                            |
|        | TS     | MPTEMP  |                                            |
| LEFT   | CCS    | MODE    | # SINCE LEFT SHIFTING IS DONE ONE PLACE AT |
|        | TCF    | GENSCL  | # A TIME, NO COMPARISON WITH 14 NEED BE    |
|        | TCF    | GENSCL  | # DONE. FOR SCALARS, SEE IF TERMINAL ROUND |
|        | TCF    | VSSL    | # DESIRED. FOR VECTORS, SHIFT IMMEDIATELY. |
| GENSCL | CS     | ADDRWD  | # PUT ROUNDING BIT (BIT 10 OF ADDRWD) INTO |
|        | EXTEND |         | # BIT 15 OF CYR WHERE THE ROUNDING BIT OF  |
|        | MP     | BIT6    | # A SHORT SHIFT LEFT WOULD BE              |
|        | TS     | CYR     |                                            |
|        | TCF    | TSSL +2 | # DO THE SHIFT.                            |

1412THE  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
1

# SCALAR DIVISION INSTRUCTIONS, DDV AND BDDV, ARE EXECUTED HERE. AT THIS POINT, THE DIVIDEND IS IN MPAC  
# AND THE DIVISOR IS IN BUF.

DDV/BDDV CS ONE # INITIALIZATION  
TS DVSIGN # +-1 FOR POSITIVE QUOTIENT -- -0 FOR NEG.  
TS DVNORMCT # DIVIDENT NORMALIZATION COUNT.  
TS MAXDVSW # NEAR-ONE DIVIDE FLAG.

CCS BUF # FORCE BUF POSITIVE WITH THE MAJOR PART  
TCF BUFPOS # NON-ZERO.  
TCF +2

TCF BUFNEG

BUFZERO TS MPAC +2 # ZERO THIS.

TC TPAGREE # FORCE SIGN AGREEMENT BEFORE OVERFLOW

CCS MPAC # TEST TO SEE IF MPAC NON-ZERO. (TOO BIG)

TCF OVFP+ # MAJOR PART OF DIVIDEND IS POSITIVE NON-0

TCF +2

TCF OVFP+ -1 # MAJOR PART OF DIVIDEND IS NEG. NON-ZERO

XCH BUF +1 # SHIFT DIVIDEND AND DIVISOR LEFT 14

XCH BUF

XCH MPAC +1

XCH MPAC

CCS BUF # TRY AGAIN ON FORMER MINOR PART.

TCF BUF+

TCF +2

TCF BUF- # OVERFLOW ON ZERO DIVISOR.

SGNDVOVF CS MPAC # SIGN OF MPAC DETERMINES SIGN OF RESULT.

EXTEND

BZMF +2

INCR DVSIGN

CAF POSMAX # NEGMAX IN MPAC PERHAPS.

TS MPAC # ON DIVISION OVERFLOW OF ANY SORT, SET

TC FINALDV +3 # SET DP MPAC TO +-POS MAX.

CAF ONE

TS OVFPND # SET OVERFLOW INDICATOR AND EXIT.

TC DANZIG

-1 INCR DVSIGN

OVFP+ CS BUF +1

TCF SGNDVOVF # LOAD LOWER ORDER PART OF DIVISOR.

BUF- EXTEND # IF BUF IS NEGATIVE, COMPLEMENT IT AND

DCS BUF # MAINTAIN DVSIGN FOR FINAL QUOTIENT SIGN.

DXCH BUF

INCR DVSIGN

# NOW -0.

```
MPAC- EXTEND # FORCE MPAC POSITIVE AS BUF IN BUF-.
```

```
EXIT IMMEDIATELY ON ZERO DIVIDEND.
```

# NOW +1 OR -0.

MPAC+

CS MPAC  
AD NEGONE# CHECK FOR DIVISION OVERFLOW. IF THE  
# MAJOR PART OF THE DIVIDEND IS LESS THAN  
# THE MAJOR PART OF THE DIVISOR BY AT  
# LEAST TWO, WE CAN PROCEED IMMEDIATELY  
# WITHOUT NORMALIZATION PRODUCING A DVMAX.

-1/2+2

OCT 60001

# USED IN SQRTSUB.

TCF +1  
CAF HALF  
DOUBLE# IF THE ABOVE DOES NOT HOLD, FORCE SIGN  
# AGREEMENT IN NUMERATOR AND DENOMINATOR  
# TO FACILITATE OVERFLOW AND NEAR-ONE  
# CHECKING.AD MPAC +1  
TS MPAC +1CAF ZERO  
AD POSMAX

ADS MPAC

CAF HALF

# SAME FOR BUF.

DOUBLE

AD BUF +1  
TS BUF +1CAF ZERO  
AD POSMAX  
ADS BUFCS MPAC  
AD BUF# CHECK MAGNITUDE OF SIGN-CORRECTED  
# OPERANDS.CCS A  
TCF DVNORM

LBUF2

ADRES BUF2

# DIVIDE OK -- WILL NOT BECOME MAXOV CASE.

TCF DVOVF

# DIVISOR NOT LESS THAN DIVIDEND -- OVF.

TS MAXDVSW

CS MPAC +1  
AD BUF +1  
EXTEND# IF THE MAJOR PARTS OF THE DIVIDEND AND  
# DIVISOR ARE EQUAL, A SPECIAL APPROXIMA-  
# TION IS USED (PROVIDED THE DIVISION IS  
# POSSIBLE, OF COURSE).BZMF DVOVF  
TCF DVNORM

# IF NO OVERFLOW.

|    |         |        |           |                                            |    |
|----|---------|--------|-----------|--------------------------------------------|----|
| 1  |         |        |           |                                            | 1  |
| 2  | BUFNORM | EXTEND |           |                                            | 2  |
| 3  |         | AUG    | DVNORMCT  | # ADD -1 TO AUGMENT SHIFT COUNT AND SHIFT  | 3  |
| 4  |         | EXTEND |           |                                            | 4  |
| 5  |         | DCA    | BUF       | # LEFT ONE PLACE.                          | 5  |
| 6  |         | DAS    | BUF       |                                            | 6  |
| 7  |         |        |           |                                            | 7  |
| 8  | DVNORM  | CA     | BUF       | # SEE IF DIVISOR NORMALIZED YET.           | 8  |
| 9  |         | DOUBLE |           |                                            | 9  |
| 10 |         | OVSF   |           |                                            | 10 |
| 11 |         | TCF    | BUFNORM   | # NO -- SHIFT LEFT ONE AND TRY AGAIN.      | 11 |
| 12 |         |        |           |                                            | 12 |
| 13 |         | DXCH   | MPAC      | # CALL DIVIDEND NORMALIZATION SEQUENCE     | 13 |
| 14 |         | INDEX  | DVNORMCT  | # PRIOR TO DOING THE DIVIDE.               | 14 |
| 15 |         | TC     | MAXTEST   |                                            | 15 |
| 16 |         |        |           |                                            | 16 |
| 17 |         | TS     | MPAC +2   | # RETURNS WITH DIVISION DONE AND C(A) = 0. | 17 |
| 18 |         | TCF    | DANZIG    |                                            | 18 |
| 19 |         |        |           |                                            | 19 |
| 20 | BUFPOS  | CCS    | A         |                                            | 20 |
| 21 |         | TCF    | BUF+      | # TO BUF+ IF BUF IS GREATER THAN +1.       | 21 |
| 22 |         |        |           |                                            | 22 |
| 23 |         | CS     | BUF +1    | # IF BUF IS +1, FORCING SIGN AGREEMENT     | 23 |
| 24 |         | EXTEND |           |                                            | 24 |
| 25 |         | BZMF   | BUF+      | # MAY CAUSE BUF TO BECOME ZERO.            | 25 |
| 26 |         |        |           | # BRANCH IF SIGNS AGREE.                   | 26 |
| 27 |         | CA     | HALF      | # SIGNS DISAGREE. FORCE AGREEMENT.         | 27 |
| 28 |         | DOUBLE |           |                                            | 28 |
| 29 | +6      | ADS    | BUF +1    |                                            | 29 |
| 30 |         | CA     | ZERO      |                                            | 30 |
| 31 |         | TS     | BUF       |                                            | 31 |
| 32 |         | TCF    | BUFZERO   |                                            | 32 |
| 33 |         |        |           |                                            | 33 |
| 34 |         |        |           |                                            | 34 |
| 35 | BUFNEG  | CCS    | A         |                                            | 35 |
| 36 |         | TCF    | BUF-      | # TO BUF- IF BUF IS LESS THAN -1.          | 36 |
| 37 |         |        |           |                                            | 37 |
| 38 |         | CA     | BUF +1    | # IF BUF IS -1, FORCING SIGN AGREEMENT     | 38 |
| 39 |         | EXTEND |           |                                            | 39 |
| 40 |         | BZMF   | BUF-      | # MAY CAUSE BUF TO BECOME ZERO.            | 40 |
| 41 |         |        |           | # BRANCH IF SIGNS AGREE.                   | 41 |
| 42 |         | CS     | HALF      | # SIGNS DISAGREE. FORCE AGREEMENT.         | 42 |
| 43 |         | TCF    | BUFPOS +6 |                                            | 43 |
| 44 |         |        |           |                                            | 44 |
| 45 |         |        |           |                                            | 45 |
| 46 |         |        |           |                                            | 46 |
| 47 |         |        |           |                                            | 47 |
| 48 |         |        |           |                                            | 48 |
| 49 |         |        |           |                                            | 49 |
| 50 |         |        |           |                                            | 50 |
| 51 |         |        |           |                                            | 51 |
| 52 |         |        |           |                                            | 52 |
| 53 |         |        |           |                                            | 53 |
| 54 |         |        |           |                                            | 54 |
| 55 |         |        |           |                                            | 55 |
| 56 |         |        |           |                                            | 56 |
| 57 |         |        |           |                                            | 57 |
| 58 |         |        |           |                                            | 58 |
| 59 |         |        |           |                                            | 59 |
| 60 |         |        |           |                                            | 60 |

# INTERPRETER

# THE FOLLOWING ARE PROLOGUES TO SHIFT THE DIVIDEND ARRIVING IN A AND L BEFORE THE DIVIDE.

|         |        |           |                                            |
|---------|--------|-----------|--------------------------------------------|
| -21D    | LXCH   | SR        | # SPECIAL PROLOGUE FOR UNIT WHEN THE       |
|         | EXTEND |           | # LENGTH OF THE ARGUMENT WAS NOT LESS THAN |
|         | MP     | HALF      | # .5. IN THIS CASE, EACH COMPONENT MUST BE |
|         | XCH    | L         | # SHIFTED RIGHT ONE TO PRODUCE A HALF-UNIT |
|         | AD     | SR        | # VECTOR.                                  |
|         | XCH    | L         |                                            |
|         | TCF    | GENDDV +1 | # WITH DP DIVIDEND IN A,L.                 |
|         | DDOUBL |           | # PROLOGUE WHICH NORMALIZES THE DIVIDEND   |
|         | DDOUBL |           | # WHEN IT IS KNOWN THAT NO DIVISION        |
|         | DDOUBL |           | # OVEFLOW WILL OCCUR.                      |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DDOUBL |           |                                            |
|         | DXCH   | MPAC      |                                            |
| MAXTEST | CCS    | MAXDVSW   | # 0 IF MAJORS MIGHT BE =, -1 OTHERWISE.    |
| BIASHI  | DEC    | .4192 B-1 | # SQRT CONSTANTS.                          |
|         | TCF    | MAXDV     | # CHECK TO SEE IF THAY ARE NOW EQUAL.      |

1412THE  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
1

```
1 # THE FOLLOWING IS A GENERAL PURPOSE DOUBLE PRECISION DIVISION ROUTINE. IT DIVIDES MPAC BY BUF AND LEAVES
2 # THE RESULT IN MPAC. THE FOLLOWING CONDITIONS MUST BE SATISFIED:
3
4 #
5 # 1. THE DIVISOR (BUF) MUST BE POSITIVE AND NOT LESS THAN .5.
6 #
7 # 2. THE DIVIDEND (MPAC) MUST BE POSITIVE WITH THE MAJOR PART OF MPAC STRICTLY LESS THAN THAT OF BUF
8 # (A SPECIAL APPROXIMATION, MAXDV, IS USED WHEN THE MAJOR PARTS ARE EQUAL).
9 #
10 # UNDERSTANDING THAT A/B = Q + S(R/B) WHERE S = 2(-14) AND Q AND R ARE QUOTIENT AND REMAINDER, RESPEC-
11 # TIVELY, THE FOLLOWING APPROXIMATION IS OBTAINED BY MULTIPLYING ABOVE AND BELOW BY C - SD AND NEGLECTING TERMS OF
12 # ORDER S-SQUARED (POSSIBLY INTRODUCING ERROR INTO THE LOW TWO BITS OF THE RESULT). SIGN AGREEMENT IS UNNECESSARY.
13 #
14 # A + SB . (R - QD) A + SB
15 # ----- = Q + S(-----) WHERE Q AND R ARE QUOTIENT AND REMAINDER OF ----- RESPECTIVELY.
16 # C + SD (C } C
17
18 GENDDV DXCH MPAC # WE NEED A AND B ONLY FOR FIRST DV.
19 +1 EXTEND
20 DV BUF # (SPECIAL UNIT PROLOGUE ENTERS HERE).
21 DXCH MPAC # A NOW CONTAINS Q AND L, R.
22
23 CS MPAC # FORM DIVIDEND FOR MINOR PART OF RESULT.
24 EXTEND
25 MP BUF +1
26 AD MPAC +1 # OVERFLOW AT THIS POINT IS POSITIVE SINCE
27 OVSK # R IS POSITIVE IN EVERY CASE.
28 TCF +5
29
30 EXTEND # OVERFLOW CAN BE REMOVED BY SUBTRACTING C
31 SU BUF # (BUF) ONCE SINCE R IS ALWAYS LESS THAN C
32 INCR MPAC # IN THIS CASE. INCR COMPENSATES SUBTRACT.
33 TCF +DOWN # (SINCE C(A) IS STILL POSITIVE).
34
35 +5 EXTEND # C(A) CAN BE MADE LESS THAN C IN MAGNI-
36 BZMF -UP # TUDE BY DIMINISHING IT BY C (SINCE C IS
37 # NOT LESS THAN .5) UNLESS C(A) = 0.
```

| 1412THE |   |
|---------|---|
| 1       |   |
| 2       |   |
| 3       |   |
| 4       |   |
| 5       |   |
| 6       |   |
| 7       |   |
| 8       |   |
| 9       |   |
| 10      |   |
| 11      |   |
| 12      |   |
| 13      |   |
| 14      |   |
| 15      |   |
| 16      |   |
| 17      |   |
| 18      |   |
| 19      |   |
| 20      |   |
| 21      |   |
| 22      |   |
| 23      |   |
| 24      |   |
| 25      |   |
| 26      |   |
| 27      |   |
| 28      |   |
| 29      |   |
| 30      |   |
| 31      |   |
| 32      |   |
| 33      |   |
| 34      |   |
| 35      |   |
| 36      |   |
| 37      |   |
| 38      |   |
| 39      |   |
| 40      |   |
| 41      |   |
| 42      |   |
| 43      |   |
| 44      |   |
| 45      |   |
| 46      |   |
| 47      |   |
| 48      |   |
| 49      |   |
| 50      |   |
| 51      |   |
| 52      |   |
| 53      |   |
| 54      |   |
| 55      |   |
| 56      |   |
| 57      |   |
| 58      |   |
| 59      |   |
| 60      |   |
| 61      |   |
| 62      |   |
| 63      |   |
| 64      |   |
| 65      |   |
| 66      |   |
| 67      |   |
| 68      |   |
| 69      |   |
| 70      |   |
| 71      |   |
| 72      |   |
| 73      |   |
| 74      |   |
| 75      |   |
| 76      |   |
| 77      | 1 |
| 78      |   |
| 79      |   |
| 80      |   |





|    |         |        |                                         |                                           |    |
|----|---------|--------|-----------------------------------------|-------------------------------------------|----|
| 1  |         |        |                                         |                                           | 1  |
| 2  | FINALDV | ZL     | # DO DV TO OBTAIN MINOR PART OF RESULT. |                                           | 2  |
| 3  |         | EXTEND |                                         |                                           | 3  |
| 4  |         | DV     | BUF                                     |                                           | 4  |
| 5  | +3      | TS     | MPAC                                    | +1                                        | 5  |
| 6  |         |        |                                         |                                           | 6  |
| 7  |         | CCS    | DVSIGN                                  | # LEAVE RESULT POSITIVE UNLESS C(DVSIGN)= | 7  |
| 8  |         | TC     | Q                                       | # -0.                                     | 8  |
| 9  |         | TC     | Q                                       |                                           | 9  |
| 10 |         | TC     | Q                                       |                                           | 10 |
| 11 |         |        |                                         |                                           | 11 |
| 12 |         | EXTEND |                                         |                                           | 12 |
| 13 |         | DCS    | MPAC                                    |                                           | 13 |
| 14 |         | DXCH   | MPAC                                    |                                           | 14 |
| 15 |         | CAF    | ZERO                                    | # SO WE ALWAYS RETURN WITH C(A) = 0.      | 15 |
| 16 |         | TC     | Q                                       |                                           | 16 |
| 17 |         |        |                                         |                                           | 17 |
| 18 |         |        |                                         |                                           | 18 |
| 19 |         |        |                                         |                                           | 19 |
| 20 |         |        |                                         |                                           | 20 |
| 21 |         |        |                                         |                                           | 21 |
| 22 |         |        |                                         |                                           | 22 |
| 23 |         |        |                                         |                                           | 23 |
| 24 |         |        |                                         |                                           | 24 |
| 25 |         |        |                                         |                                           | 25 |
| 26 |         |        |                                         |                                           | 26 |
| 27 |         |        |                                         |                                           | 27 |
| 28 |         |        |                                         |                                           | 28 |
| 29 |         |        |                                         |                                           | 29 |
| 30 |         |        |                                         |                                           | 30 |
| 31 |         |        |                                         |                                           | 31 |
| 32 |         |        |                                         |                                           | 32 |
| 33 |         |        |                                         |                                           | 33 |
| 34 |         |        |                                         |                                           | 34 |
| 35 |         |        |                                         |                                           | 35 |
| 36 |         |        |                                         |                                           | 36 |
| 37 |         |        |                                         |                                           | 37 |
| 38 |         |        |                                         |                                           | 38 |
| 39 |         |        |                                         |                                           | 39 |
| 40 |         |        |                                         |                                           | 40 |
| 41 |         |        |                                         |                                           | 41 |
| 42 |         |        |                                         |                                           | 42 |
| 43 |         |        |                                         |                                           | 43 |
| 44 |         |        |                                         |                                           | 44 |
| 45 |         |        |                                         |                                           | 45 |
| 46 |         |        |                                         |                                           | 46 |
| 47 |         |        |                                         |                                           | 47 |
| 48 |         |        |                                         |                                           | 48 |
| 49 |         |        |                                         |                                           | 49 |
| 50 |         |        |                                         |                                           | 50 |
| 51 |         |        |                                         |                                           | 51 |
| 52 |         |        |                                         |                                           | 52 |
| 53 |         |        |                                         |                                           | 53 |
| 54 |         |        |                                         |                                           | 54 |
| 55 |         |        |                                         |                                           | 55 |
| 56 |         |        |                                         |                                           | 56 |
| 57 |         |        |                                         |                                           | 57 |
| 58 |         |        |                                         |                                           | 58 |
| 59 |         |        |                                         |                                           | 59 |
| 60 |         |        |                                         |                                           | 60 |

# IF THE MAJOR PARTS OF THE DIVISOR AND DIVIDEND ARE EQUAL, BUT THE MINOR PARTS ARE SUCH THAT THE  
# DIVIDEND IS STRICTLY LESS THAN THE DIVISOR IN MAGNITUDE, THE FOLLOWING APPROXIMATION IS USED. THE ASSUMPTIONS  
# ARE THE SAME AS THE GENERAL ROUTINE WITH THE ADDITION THAT SIGN AGREEMENT IS NECESSARY (B, C, & D POSITIVE).

#  
# 
$$\frac{C + SB}{C + SD} = 37777 + S\left(\frac{(C + B - D)}{C}\right)$$
  
#

# THE DIVISION MAY BE PERFORMED IMMEDIATELY SINCE B IS STRICTLY LESS THAN D AND C IS NOT LESS THAN .5.

|       |        |          |                                               |
|-------|--------|----------|-----------------------------------------------|
| MAXDV | CS     | MPAC     | # SEE IF MAXDV CASE STILL HOLDS AFTER         |
|       | AD     | BUF      | # NORMALIZATION.                              |
|       | EXTEND |          |                                               |
|       | BZF    | +2       |                                               |
|       | TCF    | GENDDV   | # MPAC NOW LESS THAN BUFF -- DIVIDE AS USUAL. |
| +2    | CAF    | POSMAX   | # SET MAJOR PART OF RESULT.                   |
|       | TS     | MPAC     |                                               |
|       | CS     | BUF +1   | # FORM DIVIDEND OF MINOR PART OF RESULT.      |
|       | AD     | MPAC +1  |                                               |
|       | TCF    | ENDMAXDV | # GO ADD C AND DO DIVIDE, ATTACHING SIGN      |
|       |        |          | # BEFORE EXITING.                             |

# VECTOR DIVIDED BY SCALAR, V/SC, IS EXECUTED HERE. THE VECTOR IS NOW IN MPAC WITH SCALAR IN BUF.

V/SC2 CS ONE # INITIALIZE DIVIDEND NORMALIZATION COUNT  
TS DVNORMCT # AND DIVISION SIGN REGISTER.  
TS VBUF +5

TC VECAGREE # FORCE SIGN AGREEMENT IN VECTOR

DXCH BUF  
TC ALSIGNAG # SIGN AGREE BUF  
DXCH BUF

CCS BUF # FORCE DIVISOR POSITIVE WITH MAJOR PART  
TCF /BUF+ # NON-ZERO (IF POSSIBLE).  
TCF +2

TCF /BUF-

XCH BUF +1 # SHIFT VECTOR AND SCALAR LEFT 14.

XCH BUF  
XCH MPAC +1  
XCH MPAC

EXTEND # CHECK FOR OVERFLOW IN EACH CASE.  
BZF +2  
TCF DVOVF

XCH MPAC +4  
XCH MPAC +3

EXTEND  
BZF +2  
TCF DVOVF

XCH MPAC +6  
XCH MPAC +5

EXTEND  
BZF +2  
TCF DVOVF

CCS BUF  
TCF /BUF+  
TCF DVOVF # ZERO DIVISOR - OVERFLOW.  
TCF /BUF-  
TCF DVOVF

/BUF- EXTEND # ON NEGATIVE, COMPLEMENT BUF AND MAINTAIN  
DCS BUF # DVSIGN IN VBUF +5.

DXCH BUF  
INCR VBUF +5

# INTERPRETER

|    |        |        |          |                                         |    |
|----|--------|--------|----------|-----------------------------------------|----|
| 1  |        |        |          |                                         | 1  |
| 2  | /BUF+  | EXTEND |          |                                         | 2  |
| 3  |        | DCA    | BUF      | # LEAVE ABS(ORIG DIVISOR) IN BUF2       | 3  |
| 4  |        | DXCH   | BUF2     | # FOR OVERFLOW TESTING                  | 4  |
| 5  |        | TCF    | /NORM    | # NORMALIZE DIVISOR IN BUF.             | 5  |
| 6  |        |        |          |                                         | 6  |
| 7  | /NORM2 | EXTEND |          | # IF LESS THAN .5, AUGMENT DVNORMCT AND | 7  |
| 8  |        | AUG    | DVNORMCT | # DOUBLE DIVISOR.                       | 8  |
| 9  |        | EXTEND |          |                                         | 9  |
| 10 |        | DCA    | BUF      |                                         | 10 |
| 11 |        | DAS    | BUF      |                                         | 11 |
| 12 |        |        |          |                                         | 12 |
| 13 | /NORM  | CA     | BUF      | # SEE IF DIVISOR NORMALIZED.            | 13 |
| 14 |        | DOUBLE |          |                                         | 14 |
| 15 |        | OVSK   |          |                                         | 15 |
| 16 |        | TCF    | /NORM2   | # DOUBLE AND TRY AGAIN IF NOT.          | 16 |
| 17 |        |        |          |                                         | 17 |
| 18 |        | TC     | V/SCDV   | # DO X COMPONENT DIVIDE.                | 18 |
| 19 |        | DXCH   | MPAC +3  | # SUPPLY ARGUMENTS IN USUAL SEQUENCE.   | 19 |
| 20 |        | DXCH   | MPAC     |                                         | 20 |
| 21 |        | DXCH   | MPAC +3  |                                         | 21 |
| 22 |        |        |          |                                         | 22 |
| 23 |        | TC     | V/SCDV   | # Y COMPONENT.                          | 23 |
| 24 |        | DXCH   | MPAC +5  |                                         | 24 |
| 25 |        | DXCH   | MPAC     |                                         | 25 |
| 26 |        | DXCH   | MPAC +5  |                                         | 26 |
| 27 |        |        |          |                                         | 27 |
| 28 |        | TC     | V/SCDV   | # Z COMPONENT.                          | 28 |
| 29 |        | TCF    | VROTATEX | # GO RE-ARRANGE COMPONENTS BEFORE EXIT. | 29 |
| 30 |        |        |          |                                         | 30 |
| 31 |        |        |          |                                         | 31 |
| 32 |        |        |          |                                         | 32 |
| 33 |        |        |          |                                         | 33 |
| 34 |        |        |          |                                         | 34 |
| 35 |        |        |          |                                         | 35 |
| 36 |        |        |          |                                         | 36 |
| 37 |        |        |          |                                         | 37 |
| 38 |        |        |          |                                         | 38 |
| 39 |        |        |          |                                         | 39 |
| 40 |        |        |          |                                         | 40 |
| 41 |        |        |          |                                         | 41 |
| 42 |        |        |          |                                         | 42 |
| 43 |        |        |          |                                         | 43 |
| 44 |        |        |          |                                         | 44 |
| 45 |        |        |          |                                         | 45 |
| 46 |        |        |          |                                         | 46 |
| 47 |        |        |          |                                         | 47 |
| 48 |        |        |          |                                         | 48 |
| 49 |        |        |          |                                         | 49 |
| 50 |        |        |          |                                         | 50 |
| 51 |        |        |          |                                         | 51 |
| 52 |        |        |          |                                         | 52 |
| 53 |        |        |          |                                         | 53 |
| 54 |        |        |          |                                         | 54 |
| 55 |        |        |          |                                         | 55 |
| 56 |        |        |          |                                         | 56 |
| 57 |        |        |          |                                         | 57 |
| 58 |        |        |          |                                         | 58 |
| 59 |        |        |          |                                         | 59 |
| 60 |        |        |          |                                         | 60 |

# INTERPRETER

# SUBROUTINE USED BY V/SC TO DIVIDE VECTOR COMPONENT IN MPAC,+1 BY THE SCALAR GIVEN IN BUF.

V/SCDV CA VBUF +5 # REFLECTS SIGN OF SCALAR.  
TS DVSIGN

CCS MPAC # FORCE MPAC POSITIVE, EXITING ON ZERO.  
TCF /MPAC+  
TCF +2  
TCF /MPAC-

CCS MPAC +1  
TCF /MPAC+  
TC Q  
TCF /MPAC-  
TC Q

/MPAC- EXTEND # USUAL COMPLEMENTING AND SETTING OF SIGN.

DCS MPAC  
DXCH MPAC  
INCR DVSIGN

/MPAC+ CS ONE # INITIALIZE NEAR-ONE SWITCH.  
TS MAXDVSW

CS MPAC # CHECK POSSIBLE OVERFLOW.  
AD BUF2 # UNNORMALIZED INPUT DIVISOR.

CCS A  
TCF DDVCALL # NOT NEAR-ONE  
TCF +2 # +0 IS JUST POSSIBLE

TCF DVOVF # NO HOPE  
TS MAXDVSW # SIGNAL POSSIBLE NEAR-ONE CASE  
CS MPAC +1 # SEE IF DIVISION CAN BE DONE  
AD BUF2 +1

EXTEND  
BZMF DVOVF

DDVCALL DXCH MPAC # CALL PRE-DIVIDE NORMALIZATION.  
INDEX DVNORMCT

TCF MAXTEST



# THE FOLLOWING ROUTINE EXECUTES THE UNIT INSTRUCTION, WHICH TAKES THE UNIT OF THE VECTOR IN MPAC.

|        |        |          |                                            |
|--------|--------|----------|--------------------------------------------|
| UNIT   | TC     | VECAGREE | # FORCE SIGN AGREEMENT IN VECTOR           |
|        | TC     | MPACVBUF | # SAVE ARGUMENT IN VBUF                    |
|        | CAF    | ZERO     | # MUST SENSE OVERFLOW IN FOLLOWING DOT.    |
|        | XCH    | OVFIND   |                                            |
|        | TS     | TEM1     |                                            |
|        | TC     | VSQSUB   | # DOT MPAC WITH ITSELF.                    |
|        | CA     | TEM1     |                                            |
|        | XCH    | OVFIND   |                                            |
|        | EXTEND |          |                                            |
|        | BZF    | +2       |                                            |
|        | TCF    | DVOVF    |                                            |
|        | EXTEND |          |                                            |
|        | DCA    | MPAC     | # LEAVE THE SQUARE OF THE LENGTH OF THE    |
|        | INDEX  | FIXLOC   | # ARGUMENT IN LVSQUARE.                    |
|        | DXCH   | LVSQUARE |                                            |
|        | TC     | SQRTSUB  | # GO TAKE THE NORMALIZED SQUARE ROOT.      |
|        | CCS    | MPAC     | # CHECK FOR UNIT OVERFLOW.                 |
|        | TCF    | +5       | # MPAC IS NOT LESS THAN .5 UNLESS          |
|        | TS     | L        |                                            |
|        | INDEX  | FIXLOC   |                                            |
|        | DXCH   | LV       |                                            |
|        | TCF    | DVOVF    | # INPUT TO SQRTSUB WAS 0.                  |
|        | CS     | FOURTEEN | # SEE IF THE INPUT WAS SO SMALL THAT THE   |
|        | AD     | MPTMP    | # FIRST TWO REGISTERS OF THE SQUARE WERE 0 |
|        | CCS    | A        |                                            |
|        | COM    |          | # IF SO, SAVE THE NEGATIVE OF THE SHIFT    |
|        | TCF    | SMALL    | # COUNT -15D.                              |
|        | TCF    | LARGE    | # (THIS IS USUALLY THE CASE.)              |
|        | CS     | THIRTEEN | # IF THE SHIFT COUNT WAS EXACTLY 14, SET   |
|        | TS     | MPTMP    | # THE PRE-DIVIDE NORM COUNT TO -13D.       |
| SMALL2 | CA     | MPAC     | # SHIFT THE LENGTH RIGHT 14 BEFORE STORING |
|        | TS     | L        | # (SMALL EXITS TO THIS POINT).             |
|        | CAF    | ZERO     |                                            |
|        | TCF    | LARGE2   | # GO TO STORE LENGTH AND PROCEED.          |
| LARGE  | CCS    | MPTMP    | # MOST ALL CASES COME HERE.                |
|        | TCF    | LARGE3   | # SEE IF NO NORMALIZATION WAS REQUIRED BY  |
|        | CS     | SRDDV    | # SQRT, AND IF SO, SET UP FOR A SHIFT      |
|        | TS     | MPTMP    | # RIGHT 1 BEFORE DIVIDING TO PRODUCE       |
|        | EXTEND |          | # THE DESIRED HALF UNIT VECTOR.            |
|        | DCA    | MPAC     |                                            |

TCF      LARGE2



LARGE3

COM  
TS

MPTEMP

# LEAVE NEGATIVE OF SHIFT COUNT-1 FOR  
# PREDIVIDE LEFT SHIFT.COM  
INDEX

A

# PICK UP REQUIRED SHIFTING BIT TO UNNORM-  
# ALIZE THE SQRT RESULT.CAF  
TS  
EXTENDBIT14  
BUFMP  
XCH  
EXTENDMPAC +1  
BUF

# (UNNORMALIZE THE SQRT FOR LV).

MP  
XCH  
AD  
XCHMPAC  
L  
BUF  
L

LARGE2

INDEX  
DXCHFIXLOC  
LV

# LENGTH NOW STORED IN WORK AREA.

CS  
TSONE  
MAXDVSW

# NO MAXDV CASES IN UNIT.

DXCH  
DXCH  
DXCH  
TCVBUF  
MPAC  
BUF  
UNITDV# PREPARE X COMPONENT FOR DIVIDE, SETTING  
# LENGTH OF VECTOR AS DIVISOR IN BUF.DXCH  
DXCH  
DXCH  
TCVBUF +2  
MPAC  
MPAC +3  
UNITDV# DO Y AND Z IN USUAL FASHION SO WE CAN  
# EXIT THROUGH VROTATEX.DXCH  
DXCH  
DXCHVBUF +4  
MPAC  
MPAC +5TC  
TCFUNITDV  
VROTATEX

# AND EXIT.

# IF THE LENGTH OF THE ARGUMENT VECTOR WAS LESS THAN 2(-28), EACH COMPONENT MUST BE SHIFTED LEFT AT LEAST  
# 14 PLACES BEFORE THE DIVIDE. NOTE THAT IN THIS CASE, THE MAJOR PART OF EACH COMPONENT IS ZERO.

SMALL TS MPTEMP # NEGATIVE OF PRE-DIVIDE SHIFT COUNT.

CAF ZERO # SHIFT EACH COMPONENT LEFT 14.

XCH VBUF +1

XCH VBUF

XCH VBUF +3

XCH VBUF +2

XCH VBUF +5

XCH VBUF +4

CS MPTEMP

INDEX A

CAF BIT14

EXTEND

MP MPAC

TCF SMALL2

THIRTEEN = OCT15

FOURTEEN = OCT16

OCT16 = R1D1

# INTERPRETER

# THE FOLLOWING ROUTINE SETS UP THE CALL TO THE DIVIDE ROUTINES.

|        |     |        |                                            |
|--------|-----|--------|--------------------------------------------|
| UNITDV | CCS | MPAC   | # FORCE MPAC POSITIVE IF POSSIBLE, SETTING |
|        | TCF | UMPAC+ | # DVSIGN ACCORDING TO THE SIGN OF MPAC     |
|        | TCF | +2     | # SINCE THE DIVISOR IS ALWAYS POSITIVE     |
|        | TCF | UMPAC- | # HERE.                                    |

|  |     |        |                             |
|--|-----|--------|-----------------------------|
|  | CCS | MPAC   | +1                          |
|  | TCF | UMPAC+ |                             |
|  | TC  | Q      | # EXIT IMMEDIATELY ON ZERO. |
|  | TCF | UMPAC- |                             |
|  | TC  | Q      |                             |

|        |    |        |                                           |
|--------|----|--------|-------------------------------------------|
| UMPAC- | CS | ZERO   | # IF NEGATIVE, SET -0 IN DVSIGN FOR FINAL |
|        | TS | DVSIGN | # COMPLEMENT.                             |

|  |        |         |                                           |
|--|--------|---------|-------------------------------------------|
|  | EXTEND |         |                                           |
|  | DCS    | MPAC    | # PICK UP ABSOLUTE VALUE OF ARG AND JUMP. |
|  | INDEX  | MPTEMP  |                                           |
|  | TCF    | MAXTEST | -1                                        |

|        |       |         |                                     |
|--------|-------|---------|-------------------------------------|
| UMPAC+ | TS    | DVSIGN  | # SET DVSIGN FOR POSITIVE QUOTIENT. |
|        | DXCH  | MPAC    |                                     |
|        | INDEX | MPTEMP  |                                     |
|        | TCF   | MAXTEST | -1                                  |



|    |                                   |        |          |                                        |    |
|----|-----------------------------------|--------|----------|----------------------------------------|----|
| 1  | # MISCELLANEOUS UNARY OPERATIONS. |        |          |                                        | 1  |
| 2  |                                   |        |          |                                        | 2  |
| 3  |                                   |        |          |                                        | 3  |
| 4  | DSQ                               | TC     | DSQSUB   | # SQUARE THE DP CONTENTS OF MPAC.      | 4  |
| 5  |                                   | TCF    | DANZIG   |                                        | 5  |
| 6  |                                   |        |          |                                        | 6  |
| 7  | ABVALABS                          | CCS    | MODE     | # ABVAL OR ABS INSTRUCTION.            | 7  |
| 8  |                                   | TCF    | ABS      | # DO ABS ON SCALAR.                    | 8  |
| 9  |                                   | TCF    | ABS      |                                        | 9  |
| 10 |                                   |        |          |                                        | 10 |
| 11 | ABVAL                             | TC     | VSQSUB   | # DOT MPAC WITH ITSELF.                | 11 |
| 12 |                                   | LXCH   | MODE     | # MODE IS NOW DP (L ZERO AFTER DAS).   | 12 |
| 13 |                                   |        |          |                                        | 13 |
| 14 |                                   | EXTEND |          | # STORE SQUARE OF LENGTH IN WORK AREA. | 14 |
| 15 |                                   | DCA    | MPAC     |                                        | 15 |
| 16 |                                   | INDEX  | FIXLOC   |                                        | 16 |
| 17 |                                   | DXCH   | LVSQUARE |                                        | 17 |
| 18 |                                   |        |          |                                        | 18 |
| 19 |                                   |        |          |                                        | 19 |
| 20 |                                   |        |          |                                        | 20 |
| 21 |                                   |        |          |                                        | 21 |
| 22 |                                   |        |          |                                        | 22 |
| 23 |                                   |        |          |                                        | 23 |
| 24 |                                   |        |          |                                        | 24 |
| 25 |                                   |        |          |                                        | 25 |
| 26 |                                   |        |          |                                        | 26 |
| 27 |                                   |        |          |                                        | 27 |
| 28 |                                   |        |          |                                        | 28 |
| 29 |                                   |        |          |                                        | 29 |
| 30 |                                   |        |          |                                        | 30 |
| 31 |                                   |        |          |                                        | 31 |
| 32 |                                   |        |          |                                        | 32 |
| 33 |                                   |        |          |                                        | 33 |
| 34 |                                   |        |          |                                        | 34 |
| 35 |                                   |        |          |                                        | 35 |
| 36 |                                   |        |          |                                        | 36 |
| 37 |                                   |        |          |                                        | 37 |
| 38 |                                   |        |          |                                        | 38 |
| 39 |                                   |        |          |                                        | 39 |
| 40 |                                   |        |          |                                        | 40 |
| 41 |                                   |        |          |                                        | 41 |
| 42 |                                   |        |          |                                        | 42 |
| 43 |                                   |        |          |                                        | 43 |
| 44 |                                   |        |          |                                        | 44 |
| 45 |                                   |        |          |                                        | 45 |
| 46 |                                   |        |          |                                        | 46 |
| 47 |                                   |        |          |                                        | 47 |
| 48 |                                   |        |          |                                        | 48 |
| 49 |                                   |        |          |                                        | 49 |
| 50 |                                   |        |          |                                        | 50 |
| 51 |                                   |        |          |                                        | 51 |
| 52 |                                   |        |          |                                        | 52 |
| 53 |                                   |        |          |                                        | 53 |
| 54 |                                   |        |          |                                        | 54 |
| 55 |                                   |        |          |                                        | 55 |
| 56 |                                   |        |          |                                        | 56 |
| 57 |                                   |        |          |                                        | 57 |
| 58 |                                   |        |          |                                        | 58 |
| 59 |                                   |        |          |                                        | 59 |
| 60 |                                   |        |          |                                        | 60 |

# PROGRAM DESCRIPTION -- SUBROUTINE SQRT

#

# FUNCTIONAL DESCRIPTION -- DOUBLE PRECISION SQUARE ROOT ROUTINE

# THIS PROGRAM TAKES THE SQUARE ROOT OF THE 27 OR 28 MOST SIGNIFICANT BITS IN THE TRIPLE PRECISION SET OF  
# NUMBERS -- MPAC, MPAC+1, AND MPAC+2. THE ROOT IS RETURNED DOUBLE PRECISION IN MPAC AND MPAC+1.

#

# WARNING -- THIS SUBROUTINE USES A TRIPLE PRECISION INPUT. THE PROGRAMMER MUST ASSURE THE CONTENTS OF MPAC+2  
# ESPECIALLY IF THE CONTENTS OF MPAC IS SMALL OR ZERO. FOR DETAILS SEE STG MEMO NO.949.

#

# CALLING SEQUENCE -- IN INTERPRETIVE MODE, I.E., FOLLOWING `TC INTPRET', `SQRT', NO ADDRESS IS ALLOWED.

# INPUT SCALING: THE BINARY POINT IS ASSUMED TO THE RIGHT OF BIT 15. THE ANSWER IS RETURNED WITH THE SAME SCALING.

#

# SUBROUTINES -- GENSCR, MPACSHR, SQRTSUB, ABORT

#

# ABORT EXIT MODE -- ABORTS ON NEGATIVE INPUT  $-1.2 \times 10^{-4}$  (77775 OCTAL) OR LESS.

# DISPLAYS ERROR CODE 1302

# TC ABORT

# OCT 1302

#

# DEBRIS -- LOCATIONS BUF, MPTMP, ADDRWD ARE USED

SQRT

TC

SQRTSUB

# TAKE THE SQUARE ROOT OF MPAC.

CCS

MPTMP

# RETURNED NORMALIZED SQUARE ROOT. SEE IF

TCF

+2

# ANY UN-NORMALIZATION REQUIRED AND EXIT

TCF

DANZIG

# IF NOT.

AD

NEG12

# A RIGHT SHIFT OF MORE THAN 13 COULD BE

EXTEND

# REQUIRED IF INPUT WAS ZERO IN MPAC,+1.

BZMF

SQRTSHFT

# GOES HERE IN MOST CASES.

ZL

# IF A LONG SHIFT IS REQUIRED, GO TO

LXCH

ADDRWD

# GENERAL RIGHT SHIFT ROUTINES.

TCF

GENSCR +4

# ADDRWD WAS ZERO TO PREVENT ROUND.

SQRTSHFT

INDEX

MPTMP

# SELECT SHIFTING BIT AND EXIT THROUGH

CAF

BIT15

# SHIFT ROUTINES.

TS

MPTMP

CAF

ZERO

# TO ZERO MPAC +2 IN THE PROCESS.

TCF

MPACSHR +3

ABS

TC

BRANCH

# TEST SIGN OF MPAC AND COMPLEMENT IF

TCF

DANZIG

TCF

DANZIG

TCF

COMP

|       |                         |                           |                                                                                                                      |
|-------|-------------------------|---------------------------|----------------------------------------------------------------------------------------------------------------------|
| VDEF  | CS<br>ADS               | FOUR<br>PUSHLOC           | # VECTOR DEFINE -- ESSENTIALLY TREATS<br># SCALAR IN MPAC AS X COMPONENT, PUSHES UP<br># FOR Y AND THEN AGAIN FOR Z. |
|       | EXTEND<br>INDEX<br>DCA  | A<br>2                    |                                                                                                                      |
|       | DXCH<br>EXTEND<br>INDEX | MPAC<br>PUSHLOC           | +3                                                                                                                   |
|       | DCA<br>DXCH<br>TCF      | 0<br>MPAC<br>VMODE        | +5<br># MODE IS NON VECTOR.                                                                                          |
| VSQ   | TC<br>TCF               | VSQSUB<br>DMODE           | # DOT MPAC WITH ITSELF.<br># MODE IS NOW DP.                                                                         |
| PUSH  | EXTEND<br>DCA           | MPAC                      | # PUSH DOWN MPAC LEAVING IT LOADED.                                                                                  |
|       | INDEX<br>DXCH           | PUSHLOC<br>0              | # PUSH DOWN FIRST TWO REGISTERS IN EACH                                                                              |
|       | INDEX<br>CAF<br>ADS     | MODE<br>NO.WDS<br>PUSHLOC | # INCREMENT PUSHDOWN POINTER.                                                                                        |
|       | CCS<br>TCF              | MODE<br>TPUSH             | # PUSH DOWN MPAC +2.                                                                                                 |
|       | TCF                     | DANZIG                    | # DONE FOR DP.                                                                                                       |
|       | EXTEND                  |                           | # ON VECTOR, PUSH DOWN Y AND Z COMPONENTS.                                                                           |
|       | DCA<br>INDEX<br>DXCH    | MPAC<br>PUSHLOC<br>0      | +3<br>-4                                                                                                             |
|       | EXTEND<br>DCA<br>INDEX  | MPAC<br>PUSHLOC           | +5                                                                                                                   |
|       | DXCH<br>TCF             | 0<br>DANZIG               | -2                                                                                                                   |
| TPUSH | CA<br>TCF               | MPAC<br>ENDTPUSH          | +2<br>+2                                                                                                             |
| RVQ   | INDEX<br>CA<br>TS       | FIXLOC<br>QPRET<br>POLISH | # RVQ -- RETURN IVA QPRET.                                                                                           |
|       | TCF                     | GOTO                      | +4<br># (ASSUME QPRET POINTS TO FIXED ONLY.)                                                                         |

# THE FOLLOWING SUBROUTINES ARE USED IN SQUARING MPAC, IN BOTH THE SCALAR AND VECTOR SENSE. THEY ARE  
# SPECIAL CASES OF DMPSUB AND DOTSUB, PUT IN TO SAVE SOME TIME.

DSQSUB CA MPAC +1 # SQUARES THE SCALAR CONTENTS OF MPAC.

EXTEND

SQUARE

TS MPAC +2

CAF ZERO

# FORM 2(CROSS TERM).

XCH MPAC +1

EXTEND

MP MPAC

DDOUBL

# AND MAYBE OVEFLOW.

DAS MPAC +1

# AND SET A TO NET OVERFLOW.

XCH MPAC

EXTEND

SQUARE

DAS MPAC

TC Q

VSQSUB EXTEND # DOTS THE VECTOR IN MPAC WITH ITSELF.

QXCH

DOTRET

TC DSQSUB

# SQUARE THE X COMPONENT.

DXCH MPAC +3

DXCH MPAC

DXCH BUF

# SO WE CAN END IN DOTSUB.

CA MPAC +2

TS BUF +2

TC DSQSUB

# SQUARE Y COMPONENT.

DXCH MPAC +1

DAS BUF +1

AD MPAC

AD BUF

TS BUF

TCF +2

TS OVFind

# IF OVERFLOW.

DXCH MPAC +5

DXCH MPAC

TC DSQSUB

# SQUARE Z COMPONENT.

TCF ENDDOT

# END AS IN DOTSUB.

```
DOUBLE PRECISION SQUARE ROOT ROUTINE. TAKE THE SQUARE ROOT OF THE TRIPLE PRECISION (MPAC +2 USED ONLY
IN NORMALIZATION) CONTENTS OF MPAC AND LEAVE THE NORMALIZED RESULT IN MPAC (C(MPAC) GREATER THAN OR EQUAL TO
.5). THE RIGHT SHIFT COUNT (TC UNNORMALIZE) IS LEFT IN MPTMP.
```

```
SQRTSUB CAF ZERO # START BY ZEROING RIGHT SHIFT COUNT.
```

```
TS MPTMP
```

```
CCS MPAC # CHECK FOR POSITIVE ARGUMENT, SHIFTING
```

```
TCF SMPAC+ # FIRST SIGNIFICANT MPAC REGISTER INTO
```

```
TCF +2 # MPAC ITSELF.
```

```
TCF SQRTNEG # SEE IF MAG OF ARGUMENT LESS THAN 10(-4).
```

```
XCH MPAC +2 # MPAC IS ZERO -- SHIFT LEFT 14.
```

```
XCH MPAC +1
```

```
TS MPAC
```

```
CAF SEVEN
```

```
TS MPTMP # AUGMENT RIGHT SHIFT COUNTER.
```

```
CCS MPAC # SEE IF MPAC NOW PNZ.
```

```
TCF SMPAC+
```

```
TCF +2
```

```
TCF ZEROANS # NEGATIVE BUT LESS THAN 10(-4) IN MAG.
```

```
XCH MPAC +1 # XERO -- SHIFT LEFT 14 AGAIN.
```

```
TS MPAC
```

```
CAF SEVEN
```

```
ADS MPTMP # AUGMENT RIGHT SHIFT COUNTER.
```

```
CCS MPAC
```

```
TCF SMPAC+
```

```
TC Q # SQRT(0) = 0.
```

```
TCF ZEROANS
```

```
TCF FIXROOT # DO NOT LEAVE SQRTSUB WITH -0 IN MPAC.
```

```
SQRTNEG CCS A # ARGUMENT IS NEGATIVE, BUT SEE IF SIGN-
```

```
TCF SQRTABRT # CORRECTED ARGUMENT IS LESS THAN 10(-4)
```

```
CCS MPAC +1 # IN MAGNITUDE. IF SO, CALL ANSWER ZERO.
```

```
ZEROANS CAF ZERO # FORCE ANSWER TO ZERO HERE.
```

```
TCF FIXROOT
```

```
TCF SQRTABRT
```

```
TCF FIXROOT
```

```
SQRTABRT DXCH LOC
```

```
TC POOD001
```

```
OCT 1302
```



```
1 SMPAC+ AD -1/2+2 # SEE IF ARGUMENT GREATER THAN OR EQUAL TO
2 EXTEND # .5.
3
4 BZMF SRTEST # IF SO, SEE IF LESS THAN .25.
5
6 DXCH MPAC # WE WILL TAKE THE SQUARE ROOT OF MPAC/2.
7 LXCH SR # SHIFT RIGHT 1 AND GO TO THE SQRT ROUTINE
8 EXTEND
9 MP HALF
10
11 DXCH MPAC
12 XCH SR
13 ADS MPAC +1 # GUARANTEED NO OVERFLOW.
14
15 ARGHI CAF SLOPEHI # ARGUMENT BETWEEN .25 AND .5, GET A
16 EXTEND # LINEAR APPROXIMATION FOR THIS RANGE.
17 MP MPAC
18 AD BIASHI # X0/2 = (MPAC/2)(SLOPHI) + BIASHI/2.
19
20 +4 TS BUF # X0/2 (ARGLO ENTERS HERE).
21 CA MPAC # SINGLE-PRECISION THROUGHOUT.
22 ZL
23 EXTEND
24 DV BUF # (MPAC/2)/(X0/2)
25 EXTEND
26 MP HALF
27 ADS BUF # X1 = X0/2 + .5(MPAX/2)/(X0/2)
28
29 EXTEND
30 MP HALF # FORM UP X1/2.
31 DXCH MPAC # SAVE AND BRING OUT ARGUMENT.
32 EXTEND # TAKE DP QUOTIENT WITH X1.
33 DV BUF
34 TS BUF +1 # SAVE MAJOR PART OF QUOTIENT.
35 CAF ZERO # FORM MINOR PART OF QUOTIENT USING
36 XCH L # (REMAINDER,0).
37 EXTEND
38 DV BUF
39 TS L # IN PREPARATION FOR DAS.
40 CA BUF +1
41 DAS MPAC # X2 = X1/2 + (MPAC/2)X1
42 EXTEND # OVERFLOWS IF ARG. NEAR POSMAX.
43
44 BZF TCQBNK00
45 CAF POSMAX
46 TS MPAC
47 TCQBNK00 MPAC +1
48 TC Q # RETURN TO CALLER TO UNNORMALIZE, ETC.
```

SRTEST

AD  
EXTEND

QUARTER

# ARGUMENT WAS LESS THAN .5, SEE IF LESS  
# THAN .25.

BZMF

SQRTNORM

# IF SO, BEGIN NORMALIZATION.

DXCH

MPAC

# IF BETWEEN .5 AND .25, SHIFT RIGHT 1 AND

LXCH

SR

# START AT ARGLO.

EXTEND

MP

HALF

DXCH

MPAC

XCH

SR

ADS

MPAC

+1

# NO OVERFLOW.

ARGLO

CAF  
EXTEND

SLOPELO

# (NORMALIZED) ARGUMENT BETWEEN .125 AND  
# .25

MP

AD

TCF

MPAC

BIASLO

ARGHI

+4

# BEGIN SQUARE ROOT.

SQRTNM2

EXTEND

# SHIFT LEFT 2 AND INCREMENT RIGHT SHIFT

DCA

MPAC

+1

# COUNT (FOR TERMINAL UNNORMALIZATION).

DAS

MPAC

+1

AD

MPAC

ADS

MPAC

# (NO OVERFLOW).

SQRTNORM

INCR  
EXTEND

MPTEMP

# FIRST TIME THROUGH, JUST SHIFT LEFT 1

# (PUTS IN EFFECTIVE RIGHT SHIFT SINCE

DCA

MPAC

+1

# WE WANT MPAC/2).

DAS

MPAC

+1

AD

MPAC

ADS

MPAC

# (AGAIN NO OVERFLOW).

DOUBLE

TS

CYL

NORMTEST

CCS

CYL

# SEE IF ARGUMENT NOW NORMALIZED AT

CCS

CYL

# GREATER THAN .125.

TCF

SQRTNM2

# NO -- SHIFT LEFT 2 MORE AND TRY AGAIN.

TCF

ARGHI

# YES -- NOW BETWEEN .5 AND .25.

TCF

ARGLO

# ARGUMENT NOW BETWEEN .25 AND .125.

```
1 # TRIGONOMETRIC FUNCTION PACKAGE.
2 #
3 # THE FOLLOWING TRIGONOMETRIC FUNCTIONS ARE AVAILABLE AS INTERPRETIVE OPERATIONS:
4 # 1. SIN COMPUTES (1/2)SINE(2 PI MPAC).
5 # 2. COS COMPUTES (1/2)COSINE(2 PI MPAC).
6 # 3. ASIN COMPUTES (1/2PI)ARCSINE(2 MPAC).
7 # 4. ACOS COMPUTES (1/2PI)ARCCOSINE(2 MPAC).
8 #
9 # SIN-ASIN AND COS-ACOS ARE MUTUALLY INVERSE, I.E., SIN(ASIN(X)) = X.
```

```
10 COSINE TC BRANCH # FINDS COSINE USING THE IDENTITY
11 TCF +3 # COS(X) = SIN(PI/2 - ABS(X)).
```

```
12 TCF PRESINE
13 TCF PRESINE
```

```
14 +3 EXTEND
15 DCS MPAC
16 DXCH MPAC
```

```
17 PRESINE CAF QUARTER # PI/2 SCALED.
18 ADS MPAC
```

```
19 SINE DXCH MPAC # DOUBLE ARGUMENT.
20 DDOUBL
```

```
21 OVSK # SEE IF OVERFLOW PRESENT.
22 TCF +3 # IF NOT, ARGUMENT OK AS IS.
```

```
23 EXTEND # IF SO, WE LOST (OR GAINED) PI, SO
24 DCOM # COMPLEMENT MPAC USING THE IDENTITY
25 # SIN(X-(+)PI) = SIN(-X).
```

```
26 +3 DXCH MPAC
27 CA MPAC # SEE IF ARGUMENT GREATER THAN .5 IN
28 DOUBLE # MAGNITUDE. IF SO, REDUCE IT TO LESS THAN
29 TS L # .5 (+-PI/2 SCALED) AS FOLLOWS:
30 TCF SN1
```

```
31 INDEX A # IF POSITIVE, FORM PI - X, IF NEGATIVE
32 CAF NEG1/2 +1 # USE -PI -X.
```

```
33 DOUBLE
34 EXTEND
35 SU MPAC # GUARANTEED NO OVERFLOW.
36 TS MPAC
```

```
37 CS MPAC +1
38 TS MPAC +1
```



|    |     |        |                                          |                                          |    |
|----|-----|--------|------------------------------------------|------------------------------------------|----|
| 1  |     |        |                                          |                                          | 1  |
| 2  | SN1 | EXTEND | # SET UP TO EVALUATE HASTINGS POLYNOMIAL |                                          | 2  |
| 3  |     | DCA    | MPAC                                     |                                          | 3  |
| 4  |     | DXCH   | BUF2                                     |                                          | 4  |
| 5  |     | TC     | DSQSUB                                   | # SQUARE MPAC.                           | 5  |
| 6  |     |        |                                          |                                          | 6  |
| 7  |     | TC     | POLY                                     | # EVALUATE FOURTH ORDER POLYNOMIAL.      | 7  |
| 8  |     | DEC    | 3                                        |                                          | 8  |
| 9  |     | 2DEC   | +.3926990796                             |                                          | 9  |
| 10 |     |        |                                          |                                          | 10 |
| 11 |     | 2DEC   | -.6459637111                             |                                          | 11 |
| 12 |     |        |                                          |                                          | 12 |
| 13 |     | 2DEC   | +.318758717                              |                                          | 13 |
| 14 |     |        |                                          |                                          | 14 |
| 15 |     | 2DEC   | -.074780249                              |                                          | 15 |
| 16 |     |        |                                          |                                          | 16 |
| 17 |     | 2DEC   | +.009694988                              |                                          | 17 |
| 18 |     |        |                                          |                                          | 18 |
| 19 |     | CAF    | LBUF2                                    | # MULTIPLY BY ARGUMENT AND SHIFT LEFT 2. | 19 |
| 20 |     | TC     | DMPSUB -1                                |                                          | 20 |
| 21 |     |        |                                          |                                          | 21 |
| 22 |     | EXTEND |                                          |                                          | 22 |
| 23 |     | DCA    | MPAC +1                                  |                                          | 23 |
| 24 |     | DAS    | MPAC +1                                  |                                          | 24 |
| 25 |     | AD     | MPAC                                     |                                          | 25 |
| 26 |     | ADS    | MPAC                                     | # NEITHER SHIFT OVERFLOWS.               | 26 |
| 27 |     | EXTEND |                                          |                                          | 27 |
| 28 |     | DCA    | MPAC +1                                  |                                          | 28 |
| 29 |     | DAS    | MPAC +1                                  |                                          | 29 |
| 30 |     | AD     | MPAC                                     |                                          | 30 |
| 31 |     | ADS    | MPAC                                     |                                          | 31 |
| 32 |     | TCF    | DANZIG                                   |                                          | 32 |
| 33 |     |        |                                          |                                          | 33 |
| 34 |     |        |                                          |                                          | 34 |
| 35 |     |        |                                          |                                          | 35 |
| 36 |     |        |                                          |                                          | 36 |
| 37 |     |        |                                          |                                          | 37 |
| 38 |     |        |                                          |                                          | 38 |
| 39 |     |        |                                          |                                          | 39 |
| 40 |     |        |                                          |                                          | 40 |
| 41 |     |        |                                          |                                          | 41 |
| 42 |     |        |                                          |                                          | 42 |
| 43 |     |        |                                          |                                          | 43 |
| 44 |     |        |                                          |                                          | 44 |
| 45 |     |        |                                          |                                          | 45 |
| 46 |     |        |                                          |                                          | 46 |
| 47 |     |        |                                          |                                          | 47 |
| 48 |     |        |                                          |                                          | 48 |
| 49 |     |        |                                          |                                          | 49 |
| 50 |     |        |                                          |                                          | 50 |
| 51 |     |        |                                          |                                          | 51 |
| 52 |     |        |                                          |                                          | 52 |
| 53 |     |        |                                          |                                          | 53 |
| 54 |     |        |                                          |                                          | 54 |
| 55 |     |        |                                          |                                          | 55 |
| 56 |     |        |                                          |                                          | 56 |
| 57 |     |        |                                          |                                          | 57 |
| 58 |     |        |                                          |                                          | 58 |
| 59 |     |        |                                          |                                          | 59 |
| 60 |     |        |                                          |                                          | 60 |

## # ARCSIN/ARCCOS ROUTINE.

|         |                                     |                              |                                                                                                                                                                                                   |
|---------|-------------------------------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ARCSIN  | CAF<br>TCF                          | LASINEX<br>+2                | # COMPUTE ARCSIN BY USING THE IDENTITY<br># ARCSIN(X) = PI/2 - ARCCOS(X).                                                                                                                         |
| ARCCOS  | CAF<br>TS<br>TC                     | LDANZIG<br>ESCAPE<br>BRANCH  | # (EXITS IMMEDIATELY).<br># TEST SIGN OF INPUT.                                                                                                                                                   |
|         | TCF<br>TCF<br>EXTEND                | ACOSST<br>ACOSZERO           | # START IMMEDIATELY IF POSITIVE.<br># ARCCOS(0) = PI/2 = .25.<br># IF NEGATIVE, USE THE IDENTITY                                                                                                  |
|         | DCS<br>DXCH<br>CAF                  | MPAC<br>MPAC<br>TCSUBTR      | # ARCCOS(X) = PI - ARCCOS(-X), FORCING<br># ARGUMENT POSITIVE.<br># SET EXIT TO DO ABOVE BEFORE                                                                                                   |
|         | XCH<br>TS                           | ESCAPE<br>ESCAPE2            | # ARCSIN/ARCCOS CONSIDERATIONS.                                                                                                                                                                   |
| ACOSST  | CS<br>AD<br>CCS                     | HALF<br>MPAC<br>A            | # TEST MAGNITUDE OF INPUT.                                                                                                                                                                        |
|         | TCF                                 | ACOSOVF                      | # THIS IS PROBABLY AN OVERFLOW CASE.                                                                                                                                                              |
| LASINEX | TCF                                 | ASINEX                       |                                                                                                                                                                                                   |
|         | TCF                                 | ACOSST2                      | # NO OVERFLOW -- PROCEED.                                                                                                                                                                         |
|         | CCS<br>CAF<br>TCF                   | MPAC +1<br>ZERO<br>ACOS=0    | # IF MAJOR PART IS .5, CALL ANSWER 0<br># UNLESS MINOR PART NEGATIVE.                                                                                                                             |
|         | TCF                                 | ACOSST2                      |                                                                                                                                                                                                   |
| ACOS=0  | TS<br>TS<br>TC                      | MPAC +1<br>MPAC<br>ESCAPE    |                                                                                                                                                                                                   |
| ACOSST2 | EXTEND<br>DCS<br>AD<br>DXCH<br>DXCH | MPAC<br>MPAC<br>MPAC<br>BUF2 | # NOW THAT ARGUMENT IS IN PROPER RANGE,<br># BEGIN COMPUTATION. USE HASTINGS<br># APPROXIMATION ARCCOS(X) = SQRT(1-X)P(X)<br># IN A SCALED VERSION WHERE P(X) IS A<br># SEVENTH ORDER POLYNOMIAL. |
|         | TC                                  | SQRTSUB                      | # RETURNS WITH NORMALIZED SQUARE ROOT.                                                                                                                                                            |
|         | CCS<br>TCF                          | MPTMP<br>ACOSSHR             | # SEE IF UN-NORMALIZATION REQUIRED.<br># IF SO.                                                                                                                                                   |

|    |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |
| 8  |
| 9  |
| 10 |
| 11 |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 |
| 17 |
| 18 |
| 19 |
| 20 |
| 21 |
| 22 |
| 23 |
| 24 |
| 25 |
| 26 |
| 27 |
| 28 |
| 29 |
| 30 |
| 31 |
| 32 |
| 33 |
| 34 |
| 35 |
| 36 |
| 37 |
| 38 |
| 39 |
| 40 |
| 41 |
| 42 |
| 43 |
| 44 |
| 45 |
| 46 |
| 47 |
| 48 |
| 49 |
| 50 |
| 51 |
| 52 |

# INTERPRETER

|    |          |        |           |                                          |    |
|----|----------|--------|-----------|------------------------------------------|----|
| 1  |          |        |           |                                          | 1  |
| 2  | ACOSSHR  | INDEX  | A         | # THE SHIFT RIGHT IS LESS THAN 14 SINCE  | 2  |
| 3  |          | CAF    | BIT14     | # THE INPUT WAS NON-ZERO DP.             | 3  |
| 4  |          | TS     | MPTMP     |                                          | 4  |
| 5  |          | TC     | VSHRRND   | # DP SHIFT RIGHT AND ROUND.              | 5  |
| 6  |          | TCF    | ACOS3     | # PROCEED.                               | 6  |
| 7  |          |        |           |                                          | 7  |
| 8  | ACOSOVF  | EXTEND |           | # IF MAJOR PART WAS ONLY 1 MORE THAN .5, | 8  |
| 9  |          | BZF    | ACOS=0    | # CALL ANSWER ZERO.                      | 9  |
| 10 |          |        |           |                                          | 10 |
| 11 | ACOSABRT | EXTEND |           | # IF OVERFLOW, CALL ANSWER ZERO BUT      | 11 |
| 12 |          | DCA    | LOC       | # SOUND AN ALARM.                        | 12 |
| 13 |          | TC     | ALARM1    |                                          | 13 |
| 14 |          | OCT    | 1301      |                                          | 14 |
| 15 |          |        |           |                                          | 15 |
| 16 |          | CAF    | ZERO      |                                          | 16 |
| 17 |          | TCF    | ACOS=0    |                                          | 17 |
| 18 |          |        |           |                                          | 18 |
| 19 | ACOSZERO | CAF    | QUARTER   | # ACOS(0) = PI/2.                        | 19 |
| 20 |          | TCF    | ACOS=0 +1 | # SET MPAC AND EXIT VIA ESCAPE.          | 20 |
| 21 |          |        |           |                                          | 21 |
| 22 | NEG12    | DEC    | -12       |                                          | 22 |
| 23 | TCSUBTR  | TCF    | SUBTR     |                                          | 23 |
| 24 |          |        |           |                                          | 24 |
| 25 |          |        |           |                                          | 25 |
| 26 |          |        |           |                                          | 26 |
| 27 |          |        |           |                                          | 27 |
| 28 |          |        |           |                                          | 28 |
| 29 |          |        |           |                                          | 29 |
| 30 |          |        |           |                                          | 30 |
| 31 |          |        |           |                                          | 31 |
| 32 |          |        |           |                                          | 32 |
| 33 |          |        |           |                                          | 33 |
| 34 |          |        |           |                                          | 34 |
| 35 |          |        |           |                                          | 35 |
| 36 |          |        |           |                                          | 36 |
| 37 |          |        |           |                                          | 37 |
| 38 |          |        |           |                                          | 38 |
| 39 |          |        |           |                                          | 39 |
| 40 |          |        |           |                                          | 40 |
| 41 |          |        |           |                                          | 41 |
| 42 |          |        |           |                                          | 42 |
| 43 |          |        |           |                                          | 43 |
| 44 |          |        |           |                                          | 44 |
| 45 |          |        |           |                                          | 45 |
| 46 |          |        |           |                                          | 46 |
| 47 |          |        |           |                                          | 47 |
| 48 |          |        |           |                                          | 48 |
| 49 |          |        |           |                                          | 49 |
| 50 |          |        |           |                                          | 50 |
| 51 |          |        |           |                                          | 51 |
| 52 |          |        |           |                                          | 52 |
| 53 |          |        |           |                                          | 53 |
| 54 |          |        |           |                                          | 54 |
| 55 |          |        |           |                                          | 55 |
| 56 |          |        |           |                                          | 56 |
| 57 |          |        |           |                                          | 57 |
| 58 |          |        |           |                                          | 58 |
| 59 |          |        |           |                                          | 59 |
| 60 |          |        |           |                                          | 60 |

# THE FOLLOWING INSTRUCTIONS ARE AVAILABLE FOR SETTING, MODIFYING, AND BRANCHING ON INDEX REGISTERS:

- |   |     |      |                                         |
|---|-----|------|-----------------------------------------|
| # | 1.  | AXT  | ADDRESS TO INDEX TRUE.                  |
| # | 2.  | AXC  | ADDRESS TO INDEX COMPLEMENTED.          |
| # | 3.  | LXA  | LOAD INDEX FROM ERASABLE.               |
| # | 4.  | LXC  | LOAD INDEX COMPLEMENTED FROM ERASABLE.  |
| # | 5.  | SXA  | STORE INDEX IN ERASABLE.                |
| # | 6.  | XCHX | EXCHANGE INDEX REGISTER WITH ERASABLE.  |
| # | 7.  | INCR | INCREMENT INDEX REGISTER.               |
| # | 8.  | XAD  | ERASABLE ADD TO INDEX REGISTER.         |
| # | 9.  | XSU  | ERASABLE SUBTRACT FROM INDEX REGISTER.  |
| # | 10. | TIX  | BRANCH ON INDEX REGISTER AND DECREMENT. |

BANK 01

|     |        |            |                                      |
|-----|--------|------------|--------------------------------------|
| AXT | COUNT* | \$\$/INTER | # SELECT APPROPRIATE INDEX REGISTER. |
|     | TC     | TAGSUB     |                                      |
|     | CA     | POLISH     |                                      |

|        |       |          |                                     |
|--------|-------|----------|-------------------------------------|
| XSTORE | INDEX | INDEXLOC | # CONTAINS C(FIXLOC) OR C(FIXLOC)+1 |
|        | TS    | X1       |                                     |
|        | TCF   | DANZIG   |                                     |

|     |    |        |
|-----|----|--------|
| AXC | TC | TAGSUB |
|     | CS | POLISH |
|     | TC | XSTORE |

|     |       |          |                                      |
|-----|-------|----------|--------------------------------------|
| LXA | TC    | 15ADRERS | # LOAD INDEX REGISTER FROM ERASABLE. |
|     | INDEX | POLISH   |                                      |
|     | CA    | 0        |                                      |
|     | TCF   | XSTORE   |                                      |

|     |       |          |                                            |
|-----|-------|----------|--------------------------------------------|
| LXC | TC    | 15ADRERS | # LOAD NDX REG FROM ERASABLE COMPLEMENTED. |
|     | INDEX | POLISH   |                                            |
|     | CS    | 0        |                                            |
|     | TCF   | XSTORE   |                                            |

|     |       |          |                                     |
|-----|-------|----------|-------------------------------------|
| SXA | TC    | 15ADRERS | # STORE INDEX REGISTER IN ERASABLE. |
|     | INDEX | INDEXLOC |                                     |
|     | CA    | X1       |                                     |

|         |       |        |
|---------|-------|--------|
| MSTORE1 | INDEX | POLISH |
|         | TS    | 0      |
|         | TCF   | DANZIG |



|    |         |        |          |                                           |    |
|----|---------|--------|----------|-------------------------------------------|----|
| 1  |         |        |          |                                           | 1  |
| 2  | XCHX    | TC     | 15ADRERS | # EXCHANGE INDEX REGISTER WITH ERASABLE.  | 2  |
| 3  |         | INDEX  | POLISH   |                                           | 3  |
| 4  |         | CA     | 0        |                                           | 4  |
| 5  |         | INDEX  | INDEXLOC |                                           | 5  |
| 6  |         | XCH    | X1       |                                           | 6  |
| 7  |         | TCF    | MSTORE1  |                                           | 7  |
| 8  |         |        |          |                                           | 8  |
| 9  | XAD     | TC     | 15ADRERS | # ADD ERASABLE TO INDEX REGISTER.         | 9  |
| 10 |         | INDEX  | POLISH   |                                           | 10 |
| 11 |         | CA     | 0        |                                           | 11 |
| 12 | XAD2    | INDEX  | INDEXLOC |                                           | 12 |
| 13 |         | ADS    | X1       | # IGNORING OVERFLOWS.                     | 13 |
| 14 |         | TCF    | DANZIG   |                                           | 14 |
| 15 |         |        |          |                                           | 15 |
| 16 | INCR    | TC     | TAGSUB   | # INCREMENT INDEX REGISTER.               | 16 |
| 17 |         | CA     | POLISH   |                                           | 17 |
| 18 |         | TCF    | XAD2     |                                           | 18 |
| 19 |         |        |          |                                           | 19 |
| 20 | XSU     | TC     | 15ADRERS | # SUBTRACT ERASABLE FROM INDEX REGISTER.  | 20 |
| 21 |         | INDEX  | POLISH   |                                           | 21 |
| 22 |         | CS     | 0        |                                           | 22 |
| 23 |         | TCF    | XAD2     |                                           | 23 |
| 24 |         |        |          |                                           | 24 |
| 25 | TIX     | TC     | TAGSUB   | # BRANCH AND DECREMENT ON INDEX.          | 25 |
| 26 |         | INDEX  | INDEXLOC |                                           | 26 |
| 27 |         | CS     | S1       |                                           | 27 |
| 28 |         | INDEX  | INDEXLOC |                                           | 28 |
| 29 |         | AD     | X1       |                                           | 29 |
| 30 |         | EXTEND |          | # NO OPERATION IF DECREMENTED INDEX IS    | 30 |
| 31 |         | BZMF   | DANZIG   | # NEGATIVE OR ZERO.                       | 31 |
| 32 |         |        |          |                                           | 32 |
| 33 | DOTIXBR | INDEX  | INDEXLOC |                                           | 33 |
| 34 |         | XCH    | X1       | # IGNORING OVERFLOWS.                     | 34 |
| 35 |         |        |          |                                           | 35 |
| 36 |         | TCF    | GOTO     | # DO THE BRANCH USING THE CADR IN POLISH. | 36 |
| 37 |         |        |          |                                           | 37 |
| 38 |         |        |          |                                           | 38 |
| 39 |         |        |          |                                           | 39 |
| 40 |         |        |          |                                           | 40 |
| 41 |         |        |          |                                           | 41 |
| 42 |         |        |          |                                           | 42 |
| 43 |         |        |          |                                           | 43 |
| 44 |         |        |          |                                           | 44 |
| 45 |         |        |          |                                           | 45 |
| 46 |         |        |          |                                           | 46 |
| 47 |         |        |          |                                           | 47 |
| 48 |         |        |          |                                           | 48 |
| 49 |         |        |          |                                           | 49 |
| 50 |         |        |          |                                           | 50 |
| 51 |         |        |          |                                           | 51 |
| 52 |         |        |          |                                           | 52 |
| 53 |         |        |          |                                           | 53 |
| 54 |         |        |          |                                           | 54 |
| 55 |         |        |          |                                           | 55 |
| 56 |         |        |          |                                           | 56 |
| 57 |         |        |          |                                           | 57 |
| 58 |         |        |          |                                           | 58 |
| 59 |         |        |          |                                           | 59 |
| 60 |         |        |          |                                           | 60 |

# INTERPRETER

|    |                                                                                                       |      |          |                                            |    |
|----|-------------------------------------------------------------------------------------------------------|------|----------|--------------------------------------------|----|
| 1  | # SUBROUTINE TO CONVERT AN ERASABLE ADDRESS (11 BITS) TO AN EBANK SETTING AND SUBADDRESS.             |      |          |                                            | 1  |
| 2  |                                                                                                       |      |          |                                            | 2  |
| 3  |                                                                                                       |      |          |                                            | 3  |
| 4  | 15ADRERS                                                                                              | CS   | POLISH   |                                            | 4  |
| 5  |                                                                                                       | AD   | DEC45    |                                            | 5  |
| 6  |                                                                                                       | CCS  | A        | # DOES THE ADDRESS POINT TO THE WORK AREA? | 6  |
| 7  |                                                                                                       | CA   | FIXLOC   | # YES. ADD FIXLOC. EBANK OK AS IS.         | 7  |
| 8  |                                                                                                       | TCF  | +5       |                                            | 8  |
| 9  |                                                                                                       |      |          |                                            | 9  |
| 10 |                                                                                                       | CA   | OCT1400  | # NO. SET EBANK & MAKE UP SUBADDRESS.      | 10 |
| 11 |                                                                                                       | XCH  | POLISH   |                                            | 11 |
| 12 |                                                                                                       | TS   | EBANK    |                                            | 12 |
| 13 |                                                                                                       | MASK | LOW8     |                                            | 13 |
| 14 | +5                                                                                                    | ADS  | POLISH   | # FALL INTO TAGSUB, AND RETURN VIA Q.      | 14 |
| 15 |                                                                                                       |      |          |                                            | 15 |
| 16 | # SUBROUTINE WHICH SETS THE ADDRESS OF THE SPECIFIED INDEX IN INDEXLOC. (ACTUALLY, THE ADDRESS -38D.) |      |          |                                            | 16 |
| 17 |                                                                                                       |      |          |                                            | 17 |
| 18 | TAGSUB                                                                                                | CA   | FIXLOC   |                                            | 18 |
| 19 |                                                                                                       | TS   | INDEXLOC |                                            | 19 |
| 20 |                                                                                                       |      |          |                                            | 20 |
| 21 |                                                                                                       | CCS  | CYR      | # BIT 15 SPECIFIES INDEX.                  | 21 |
| 22 |                                                                                                       | INCR | INDEXLOC | # 0 MEANS USE X2.                          | 22 |
| 23 |                                                                                                       | TC   | Q        |                                            | 23 |
| 24 |                                                                                                       | TC   | Q        | # 1 FOR X1.                                | 24 |
| 25 |                                                                                                       |      |          |                                            | 25 |
| 26 |                                                                                                       |      |          |                                            | 26 |
| 27 |                                                                                                       |      |          |                                            | 27 |
| 28 |                                                                                                       |      |          |                                            | 28 |
| 29 |                                                                                                       |      |          |                                            | 29 |
| 30 |                                                                                                       |      |          |                                            | 30 |
| 31 |                                                                                                       |      |          |                                            | 31 |
| 32 |                                                                                                       |      |          |                                            | 32 |
| 33 |                                                                                                       |      |          |                                            | 33 |
| 34 |                                                                                                       |      |          |                                            | 34 |
| 35 |                                                                                                       |      |          |                                            | 35 |
| 36 |                                                                                                       |      |          |                                            | 36 |
| 37 |                                                                                                       |      |          |                                            | 37 |
| 38 |                                                                                                       |      |          |                                            | 38 |
| 39 |                                                                                                       |      |          |                                            | 39 |
| 40 |                                                                                                       |      |          |                                            | 40 |
| 41 |                                                                                                       |      |          |                                            | 41 |
| 42 |                                                                                                       |      |          |                                            | 42 |
| 43 |                                                                                                       |      |          |                                            | 43 |
| 44 |                                                                                                       |      |          |                                            | 44 |
| 45 |                                                                                                       |      |          |                                            | 45 |
| 46 |                                                                                                       |      |          |                                            | 46 |
| 47 |                                                                                                       |      |          |                                            | 47 |
| 48 |                                                                                                       |      |          |                                            | 48 |
| 49 |                                                                                                       |      |          |                                            | 49 |
| 50 |                                                                                                       |      |          |                                            | 50 |
| 51 |                                                                                                       |      |          |                                            | 51 |
| 52 |                                                                                                       |      |          |                                            | 52 |
| 53 |                                                                                                       |      |          |                                            | 53 |
| 54 |                                                                                                       |      |          |                                            | 54 |
| 55 |                                                                                                       |      |          |                                            | 55 |
| 56 |                                                                                                       |      |          |                                            | 56 |
| 57 |                                                                                                       |      |          |                                            | 57 |
| 58 |                                                                                                       |      |          |                                            | 58 |
| 59 |                                                                                                       |      |          |                                            | 59 |
| 60 |                                                                                                       |      |          |                                            | 60 |

1412THE

1

|    |                                                                           |     |           |                                                              |    |
|----|---------------------------------------------------------------------------|-----|-----------|--------------------------------------------------------------|----|
| 1  |                                                                           |     |           |                                                              | 1  |
| 2  | # MISCELLANEOUS OPERATION CODES WITH DIRECT ADDRESSES. INCLUDED HERE ARE: |     |           |                                                              | 2  |
| 3  | #                                                                         | 1.  | ITA       | STORE CPRET (RETURN ADDRESS) IN ERASABLE.                    | 3  |
| 4  | #                                                                         | 2.  | CALL      | CALL A SUBROUTINE, LEAVING RETURN IN QPRET.                  | 4  |
| 5  | #                                                                         | 3.  | RTB       | RETURN TO BASIC LANGUAGE AT THE GIVEN ADDRESS.               | 5  |
| 6  | #                                                                         | 4.  | BHIZ      | BRANCH IF THE HIGH ORDER OF MPAC IS ZERO (SINGLE PRECISION). | 6  |
| 7  | #                                                                         | 5.  | BOV       | BRANCH ON OVERFLOW.                                          | 7  |
| 8  | #                                                                         | 6.  | GOTO      | SIMPLE SEQUENCE CHANGE.                                      | 8  |
| 9  |                                                                           |     |           |                                                              | 9  |
| 10 | RTB/BHIZ                                                                  | CCS | CYR       |                                                              | 10 |
| 11 | RTB                                                                       | CA  | POLISH    |                                                              | 11 |
| 12 |                                                                           | TC  | SWCALL -1 | # SO A "TC Q" FROM ROUTINE LEADS TO DANZIG                   | 12 |
| 13 |                                                                           |     |           |                                                              | 13 |
| 14 | BHIZ                                                                      | CCS | MPAC      |                                                              | 14 |
| 15 |                                                                           | TCF | DANZIG    |                                                              | 15 |
| 16 |                                                                           | TCF | GOTO      |                                                              | 16 |
| 17 |                                                                           | TCF | DANZIG    |                                                              | 17 |
| 18 |                                                                           | TCF | GOTO      |                                                              | 18 |
| 19 |                                                                           |     |           |                                                              | 19 |
| 20 | BOV(B)                                                                    | CCS | OVFIND    | # BRANCH ON OVERFLOW TO BASIC OR INTERP.                     | 20 |
| 21 |                                                                           | TCF | +2        |                                                              | 21 |
| 22 |                                                                           | TCF | DANZIG    |                                                              | 22 |
| 23 |                                                                           | TS  | OVFIND    |                                                              | 23 |
| 24 |                                                                           | CCS | CYR       |                                                              | 24 |
| 25 |                                                                           | TCF | RTB       | # IF BASIC.                                                  | 25 |
| 26 | B5TOBB                                                                    | OCT | 360       |                                                              | 26 |
| 27 |                                                                           | TCF | GOTO      |                                                              | 27 |
| 28 |                                                                           |     |           |                                                              | 28 |
| 29 |                                                                           |     |           |                                                              | 29 |
| 30 |                                                                           |     |           |                                                              | 30 |
| 31 |                                                                           |     |           |                                                              | 31 |
| 32 |                                                                           |     |           |                                                              | 32 |
| 33 |                                                                           |     |           |                                                              | 33 |
| 34 |                                                                           |     |           |                                                              | 34 |
| 35 |                                                                           |     |           |                                                              | 35 |
| 36 |                                                                           |     |           |                                                              | 36 |
| 37 |                                                                           |     |           |                                                              | 37 |
| 38 |                                                                           |     |           |                                                              | 38 |
| 39 |                                                                           |     |           |                                                              | 39 |
| 40 |                                                                           |     |           |                                                              | 40 |
| 41 |                                                                           |     |           |                                                              | 41 |
| 42 |                                                                           |     |           |                                                              | 42 |
| 43 |                                                                           |     |           |                                                              | 43 |
| 44 |                                                                           |     |           |                                                              | 44 |
| 45 |                                                                           |     |           |                                                              | 45 |
| 46 |                                                                           |     |           |                                                              | 46 |
| 47 |                                                                           |     |           |                                                              | 47 |
| 48 |                                                                           |     |           |                                                              | 48 |
| 49 |                                                                           |     |           |                                                              | 49 |
| 50 |                                                                           |     |           |                                                              | 50 |
| 51 |                                                                           |     |           |                                                              | 51 |
| 52 |                                                                           |     |           |                                                              | 52 |
| 53 |                                                                           |     |           |                                                              | 53 |
| 54 |                                                                           |     |           |                                                              | 54 |
| 55 |                                                                           |     |           |                                                              | 55 |
| 56 |                                                                           |     |           |                                                              | 56 |
| 57 |                                                                           |     |           |                                                              | 57 |
| 58 |                                                                           |     |           |                                                              | 58 |
| 59 |                                                                           |     |           |                                                              | 59 |
| 60 |                                                                           |     |           |                                                              | 60 |

|    |          |       |          |                                            |    |
|----|----------|-------|----------|--------------------------------------------|----|
| 1  |          |       |          |                                            | 1  |
| 2  | BZE/GOTO | CCS   | CYR      | # SEE WHICH OP-CODE IS DESIRED.            | 2  |
| 3  |          | TC    | BRANCH   | # DO BZE.                                  | 3  |
| 4  |          | TCF   | DANZIG   |                                            | 4  |
| 5  |          | TCF   | GOTO     | # DO GOTO.                                 | 5  |
| 6  |          | TCF   | DANZIG   |                                            | 6  |
| 7  |          |       |          |                                            | 7  |
| 8  | BPL/BMN  | CCS   | CYR      |                                            | 8  |
| 9  |          | TCF   | BPL      |                                            | 9  |
| 10 | 5B10     | DEC   | 5 B+10   | # SHIFTS OP CODE IN SWITCH INSTRUCTION ADR | 10 |
| 11 |          |       |          |                                            | 11 |
| 12 |          | TC    | BRANCH   | # DO BMN                                   | 12 |
| 13 |          | TCF   | DANZIG   |                                            | 13 |
| 14 |          | TCF   | DANZIG   |                                            | 14 |
| 15 |          | TCF   | GOTO     | # ONLY IF NNZ.                             | 15 |
| 16 |          |       |          |                                            | 16 |
| 17 | BPL      | TC    | BRANCH   |                                            | 17 |
| 18 |          | TCF   | GOTO     | # IF POSITIVE OR ZERO.                     | 18 |
| 19 |          | TCF   | GOTO     |                                            | 19 |
| 20 |          | TCF   | DANZIG   |                                            | 20 |
| 21 |          |       |          |                                            | 21 |
| 22 | CALL/ITA | CCS   | CYR      |                                            | 22 |
| 23 |          | TCF   | CALL     |                                            | 23 |
| 24 |          |       |          |                                            | 24 |
| 25 |          | TC    | CCSHOLE  |                                            | 25 |
| 26 |          | TC    | 15ADRERS | # STORE QPRET. (TAGSUB AFTER 15ADRERS IS   | 26 |
| 27 |          | INDEX | FIXLOC   | # SLOW IN THIS CASE, BUT SAVES STORAGE.)   | 27 |
| 28 |          | CA    | QPRET    |                                            | 28 |
| 29 |          | TCF   | MSTORE1  |                                            | 29 |
| 30 |          |       |          |                                            | 30 |
| 31 |          |       |          |                                            | 31 |
| 32 |          |       |          |                                            | 32 |
| 33 |          |       |          |                                            | 33 |
| 34 |          |       |          |                                            | 34 |
| 35 |          |       |          |                                            | 35 |
| 36 |          |       |          |                                            | 36 |
| 37 |          |       |          |                                            | 37 |
| 38 |          |       |          |                                            | 38 |
| 39 |          |       |          |                                            | 39 |
| 40 |          |       |          |                                            | 40 |
| 41 |          |       |          |                                            | 41 |
| 42 |          |       |          |                                            | 42 |
| 43 |          |       |          |                                            | 43 |
| 44 |          |       |          |                                            | 44 |
| 45 |          |       |          |                                            | 45 |
| 46 |          |       |          |                                            | 46 |
| 47 |          |       |          |                                            | 47 |
| 48 |          |       |          |                                            | 48 |
| 49 |          |       |          |                                            | 49 |
| 50 |          |       |          |                                            | 50 |
| 51 |          |       |          |                                            | 51 |
| 52 |          |       |          |                                            | 52 |
| 53 |          |       |          |                                            | 53 |
| 54 |          |       |          |                                            | 54 |
| 55 |          |       |          |                                            | 55 |
| 56 |          |       |          |                                            | 56 |
| 57 |          |       |          |                                            | 57 |
| 58 |          |       |          |                                            | 58 |
| 59 |          |       |          |                                            | 59 |
| 60 |          |       |          |                                            | 60 |

# INTERPRETER

# THE FOLLOWING OPERATIONS ARE AVAILABLE FOR ALTERING AND TESTING INTERPRETATIVE SWITCHES:

|   |    |        |                                          |
|---|----|--------|------------------------------------------|
| # | 00 | BONSET | SET A SWITCH AND DO A GOTO IF IT WAS ON. |
| # | 01 | SETGO  | SET A SWITCH AND DO A GOTO.              |
| # | 02 | BOFSET | SET A SWITCH AND DOA GOTO IF IT WAS OFF  |
| # | 03 | SET    | SET A SWITCH.                            |
| # | 04 | BONINV | INVERT A SWITCH AND BRANCH IF IT WAS ON. |
| # | 05 | INVGO  | INVERT A SWITCH AND DO A GOTO.           |
| # | 06 | BOFINV | INVERT A SWITCH AND BRANCH IF IT WAS OFF |
| # | 07 | INVERT | INVERT A SWITCH.                         |
| # | 10 | BONCLR | CLEAR A SWITCH AND BRANCH IF IT WAS ON.  |
| # | 11 | CLRGO  | CLEAR A SWITCH AND DO A GOTO.            |
| # | 12 | BOFCLR | CLEAR A SWITCH AND BRANCH IF IT WAS OFF. |
| # | 13 | CLEAR  | CLEAR A SWITCH.                          |
| # | 14 | BON    | BRANCH IF A SWITCH WAS ON.               |
| # | 16 | BOFF   | BRANCH IF A SWITCH WAS OFF.              |

# THE ADDRESS SUPPLIED WITH THE SWITCH INSTRUCTION IS INTERPRETED AS FOLLOWS:

|   |          |                                             |
|---|----------|---------------------------------------------|
| # | BITS 1-4 | SWITCH BIT NUMBER (1-15).                   |
| # | BITS 5-8 | SWITCH OPERATION NUMBER                     |
| # | BITS 9-  | SWITCH WORD NUMBER (UP TO 64 SWITCH WORDS). |

# THE ADDRESS ITSELF IS MADE UP BY THE YUL SYSTEM ASSEMBLER. THE BRANCH INSTRUCTIONS REQUIRE TWO  
# ADDRESSES, THE SECOND TAKEN AS THE DIRECT (OR INDIRECT IF IN ERASABLE) ADDRESS OF THE BRANCH.

|          |        |        |                                            |
|----------|--------|--------|--------------------------------------------|
| SWITCHES | CAF    | LOW4   | # LEAVE THE SWITCH BIT IN SWBIT.           |
|          | MASK   | POLISH |                                            |
|          | INDEX  | A      |                                            |
|          | CAF    | BIT15  | # (NUMBER FROM LEFT TO RIGHT.)             |
|          | TS     | SWBIT  |                                            |
|          | CAF    | BIT7   | # LEAVE THE SWITCH NUMBER IN SWWORD.       |
|          | EXTEND |        |                                            |
|          | MP     | POLISH |                                            |
|          | TS     | SWWORD |                                            |
|          | INHINT |        | # DURING SWITCH CHANGE SO RUPT CAN USE TOO |
|          | INDEX  | A      | # LEAVE THE SWITCH WORD ITSELF IN L.       |
|          | CA     | STATE  |                                            |
|          | TS     | Q      | # Q WILL BE USED AS A CHANNEL.             |

| #  |         | INTERPRETER | OPERATION | OPERATION AS IN BITS                       |
|----|---------|-------------|-----------|--------------------------------------------|
| 2  |         | CAF         | BIT11     |                                            |
| 3  |         | EXTEND      |           | # DISPATCH SWITCH BIT OPERATION AS IN BITS |
| 4  |         | MP          | POLISH    | # 7-8 OF POLISH.                           |
| 5  |         | MASK        | B3TOB4    | # GETS 4X2-BIT CODE.                       |
| 6  |         | INDEX       | A         |                                            |
| 7  |         | TCF         | +1        |                                            |
| 8  |         |             |           |                                            |
| 9  | +1      | CA          | SWBIT     | # 00 -- SET SWITCH IN QUESTION.            |
| 10 |         | EXTEND      |           |                                            |
| 11 |         | ROR         | QCHAN     |                                            |
| 12 |         | TCF         | SWSTORE   |                                            |
| 13 |         |             |           |                                            |
| 14 | +5      | CA          | SWBIT     | # 01 -- INVERT SWITCH.                     |
| 15 |         | EXTEND      |           |                                            |
| 16 |         | RXOR        | QCHAN     |                                            |
| 17 |         | TCF         | SWSTORE   |                                            |
| 18 |         |             |           |                                            |
| 19 | +9D     | CS          | SWBIT     | # 10 -- CLEAR.                             |
| 20 |         | MASK        | Q         |                                            |
| 21 | SWSTORE | INDEX       | SWWORD    |                                            |
| 22 |         | TS          | STATE     | # NEW SWITCH WORD.                         |
| 23 |         |             |           |                                            |
| 24 |         |             |           |                                            |
| 25 |         |             |           |                                            |
| 26 |         |             |           |                                            |
| 27 |         |             |           |                                            |
| 28 |         |             |           |                                            |
| 29 |         |             |           |                                            |
| 30 |         |             |           |                                            |
| 31 |         |             |           |                                            |
| 32 |         |             |           |                                            |
| 33 |         |             |           |                                            |
| 34 |         |             |           |                                            |
| 35 |         |             |           |                                            |
| 36 |         |             |           |                                            |
| 37 |         |             |           |                                            |
| 38 |         |             |           |                                            |
| 39 |         |             |           |                                            |
| 40 |         |             |           |                                            |
| 41 |         |             |           |                                            |
| 42 |         |             |           |                                            |
| 43 |         |             |           |                                            |
| 44 |         |             |           |                                            |
| 45 |         |             |           |                                            |
| 46 |         |             |           |                                            |
| 47 |         |             |           |                                            |
| 48 |         |             |           |                                            |
| 49 |         |             |           |                                            |
| 50 |         |             |           |                                            |
| 51 |         |             |           |                                            |
| 52 |         |             |           |                                            |
| 53 |         |             |           |                                            |
| 54 |         |             |           |                                            |
| 55 |         |             |           |                                            |
| 56 |         |             |           |                                            |
| 57 |         |             |           |                                            |
| 58 |         |             |           |                                            |
| 59 |         |             |           |                                            |
| 60 |         |             |           |                                            |



|    |        |        |          |                                          |    |
|----|--------|--------|----------|------------------------------------------|----|
| 1  |        |        |          |                                          | 1  |
| 2  | +13D   | RELINT |          | # 11 -- NOOP.                            | 2  |
| 3  |        | CAF    | BIT13    |                                          | 3  |
| 4  |        | EXTEND |          | # DISPATCH SEQUENCE CHANGING OR BRANCING | 4  |
| 5  |        | MP     | POLISH   | # CODE.                                  | 5  |
| 6  |        | MASK   | B3TOB4   |                                          | 6  |
| 7  |        | INDEX  | A        |                                          | 7  |
| 8  |        | TCF    | +1       | # ORIGINALLY STORED IN BITS 5-6          | 8  |
| 9  |        |        |          |                                          | 9  |
| 10 | +1     | CS     | Q        | # 00 -- BRANCH IF ON.                    | 10 |
| 11 | TEST   | MASK   | SWBIT    |                                          | 11 |
| 12 |        | CCS    | A        |                                          | 12 |
| 13 |        | TCF    | SWSKIP   |                                          | 13 |
| 14 |        |        |          |                                          | 14 |
| 15 | +5     | TCF    | SWBRANCH | # 01 -- GO TO.                           | 15 |
| 16 |        |        |          |                                          | 16 |
| 17 |        | TCF    | SWSKIP   | # HERE ONLY ON BIT 15.                   | 17 |
| 18 |        |        |          |                                          | 18 |
| 19 |        | TC     | CCSHOLE  |                                          | 19 |
| 20 |        | TC     | CCSHOLE  |                                          | 20 |
| 21 |        |        |          |                                          | 21 |
| 22 | +9D    | CA     | Q        | # 10 -- BRANCH IF OFF.                   | 22 |
| 23 |        | TCF    | TEST     |                                          | 23 |
| 24 |        |        |          |                                          | 24 |
| 25 | B3TOB4 | OCT    | 0014     |                                          | 25 |
| 26 | SWSKIP | INCR   | LOC      |                                          | 26 |
| 27 |        |        |          |                                          | 27 |
| 28 | SW/    | EQUALS | SWITCHES |                                          | 28 |
| 29 |        |        |          |                                          | 29 |
| 30 | +13D   | TCF    | DANZIG   | # 11 -- NOOP.                            | 30 |
| 31 |        |        |          |                                          | 31 |
| 32 |        |        |          |                                          | 32 |
| 33 |        |        |          |                                          | 33 |
| 34 |        |        |          |                                          | 34 |
| 35 |        |        |          |                                          | 35 |
| 36 |        |        |          |                                          | 36 |
| 37 |        |        |          |                                          | 37 |
| 38 |        |        |          |                                          | 38 |
| 39 |        |        |          |                                          | 39 |
| 40 |        |        |          |                                          | 40 |
| 41 |        |        |          |                                          | 41 |
| 42 |        |        |          |                                          | 42 |
| 43 |        |        |          |                                          | 43 |
| 44 |        |        |          |                                          | 44 |
| 45 |        |        |          |                                          | 45 |
| 46 |        |        |          |                                          | 46 |
| 47 |        |        |          |                                          | 47 |
| 48 |        |        |          |                                          | 48 |
| 49 |        |        |          |                                          | 49 |
| 50 |        |        |          |                                          | 50 |
| 51 |        |        |          |                                          | 51 |
| 52 |        |        |          |                                          | 52 |
| 53 |        |        |          |                                          | 53 |
| 54 |        |        |          |                                          | 54 |
| 55 |        |        |          |                                          | 55 |
| 56 |        |        |          |                                          | 56 |
| 57 |        |        |          |                                          | 57 |
| 58 |        |        |          |                                          | 58 |
| 59 |        |        |          |                                          | 59 |
| 60 |        |        |          |                                          | 60 |

BLOCK 02

COUNT\* \$\$/FCONS

# THE FOLLOWING TABLE OF 18 VALUES IS INDEXED. DO NOT INSERT OR REMOVE ANY QUANTITIES

|         |     |       |                       |
|---------|-----|-------|-----------------------|
| DPOSMAX | OCT | 37777 | # MUST PRECEDE POSMAX |
| POSMAX  | OCT | 37777 |                       |

LIMITS = NEG1/2

|        |     |        |                                                                      |
|--------|-----|--------|----------------------------------------------------------------------|
| NEG1/2 | OCT | -20000 | # USED BY SIN ROUTINE (MUST BE TWO<br># LOCATIONS IN FRONT OF BIT14) |
|--------|-----|--------|----------------------------------------------------------------------|

# BIT TABLE

|       |     |       |
|-------|-----|-------|
| BIT15 | OCT | 40000 |
| BIT14 | OCT | 20000 |
| BIT13 | OCT | 10000 |
| BIT12 | OCT | 04000 |
| BIT11 | OCT | 02000 |
| BIT10 | OCT | 01000 |
| BIT9  | OCT | 00400 |
| BIT8  | OCT | 00200 |
| BIT7  | OCT | 00100 |
| BIT6  | OCT | 00040 |
| BIT5  | OCT | 00020 |
| BIT4  | OCT | 00010 |
| BIT3  | OCT | 00004 |
| BIT2  | OCT | 00002 |
| BIT1  | OCT | 00001 |

# DO NOT DESTROY THIS COMBINATION, SINCE IT IS USED IN DOUBLE PRECISION INSTRUCTIONS.

|        |     |       |                     |
|--------|-----|-------|---------------------|
| NEGO   | OCT | -0    | # MUST PRECEDE ZERO |
| ZERO   | OCT | 0     | # MUST FOLLOW NEGO  |
| # BIT1 | OCT | 00001 |                     |

|          |     |   |               |
|----------|-----|---|---------------|
| # NO.WDS | OCT | 2 | # INTERPRETER |
| # OCTAL3 | OCT | 3 | # INTERPRETER |
| # R3D1   | OCT | 4 | # PINBALL     |

|          |     |   |               |
|----------|-----|---|---------------|
| FIVE     | OCT | 5 |               |
| # REVCNT | OCT | 6 | # INTERPRETER |
| SEVEN    | OCT | 7 |               |

|        |     |       |           |
|--------|-----|-------|-----------|
| # BIT4 | OCT | 00010 |           |
| # R2D1 | OCT | 11    | # PINBALL |
| OCT11  | =   | R2D1  | # P20S    |

|          |     |    |                            |
|----------|-----|----|----------------------------|
| # BINCON | DEC | 10 | # PINBALL (OCTAL 12)       |
| ELEVEN   | DEC | 11 |                            |
| # OCT14  | OCT | 14 | # ALARM AND ABORT (FILLER) |

|        |     |    |           |
|--------|-----|----|-----------|
| OCT15  | OCT | 15 |           |
| # R1D1 | OCT | 16 | # PINBALL |
| LOW4   | OCT | 17 |           |



|           |     |          |                                                        |
|-----------|-----|----------|--------------------------------------------------------|
| # BIT5    | OCT | 00020    |                                                        |
| # ND1     | OCT | 21       | # PINBALL                                              |
| # VD1     | OCT | 23       | # PINBALL                                              |
| # OCT24   | OCT | 24       | # SERVICE ROUTINES                                     |
| # MD1     | OCT | 25       | # PINBALL                                              |
| BITS4&5   | OCT | 30       |                                                        |
| # OCT31   | OCT | 31       | # SERVICE ROUTINES                                     |
| OCT33     | OCT | 33       |                                                        |
| DEC27     | =   | OCT33    |                                                        |
| OCT35     | OCT | 35       |                                                        |
| DEC29     | =   | OCT35    |                                                        |
| CALLCODE  | OCT | 00032    |                                                        |
| # LOW5    | OCT | 37       | # PINBALL                                              |
| # 33DEC   | DEC | 33       | # PINBALL (OCTAL 41)                                   |
| # 34DEC   | DEC | 34       | # PINBALL (OCTAL 42)                                   |
| TBUILDFX  | DEC | 37       | # BUILDUP FOR CONVIENCE IN DAPTESTING                  |
| TDECAYFX  | DEC | 38       | # CONVENIENCE FOR DAPTESTING                           |
| # BIT6    | OCT | 00040    |                                                        |
| OCT50     | OCT | 50       |                                                        |
| DEC45     | DEC | 45       |                                                        |
| SUPER011  | OCT | 60       | # BITS FOR SUPERBNK SETTING 011.                       |
| .5SEC     | DEC | 50       |                                                        |
| # BIT7    | OCT | 00100    |                                                        |
| SUPER100  | =   | BIT7     | # BITS FOR SUPERBNK SETTING 100<br># (LAST 4K OF ROPE) |
| SUPER101  | OCT | 120      | # BITS FOR SUPERBNK SETTING 101                        |
| # OCT121  | OCT | 121      | # SERVICE ROUTINES<br># (FIRST 8K OF ACM)              |
| SUPER110  | OCT | 140      | # BITS FOR SUPERBNK SETTING 110.<br># (LAST 8K OF ACM) |
| 1SEC      | DEC | 100      |                                                        |
| # LOW7    | OCT | 177      | # INTERPRETER                                          |
| # BIT8    | OCT | 00200    |                                                        |
| # OT215   | OCT | 215      | # ALARM AND ABORT                                      |
| # 8,5     | OCT | 00220    | # P20-P25 SUNDANCE                                     |
| 2SECS     | DEC | 200      |                                                        |
| # LOW8    | OCT | 377      | # PINBALL                                              |
| # BIT9    | OCT | 00400    |                                                        |
| GN/CCODE  | OCT | 00401    | # SET S/C CONTROL SWITCH TO G/N                        |
| 3SECS     | DEC | 300      |                                                        |
| 4SECS     | DEC | 400      |                                                        |
| LOW9      | OCT | 777      |                                                        |
| # BIT10   | OCT | 01000    |                                                        |
| # 5.5DEGS | DEC | .03056   | # P20-P25 SUNDANCE (OCTAL 00765)                       |
| # OCT1103 | OCT | 1103     | # ALARM AND ABORT                                      |
| C5/2      | DEC | .0363551 | # (OCTAL 01124)                                        |
| V05N09    | VN  | 0509     | # (SAME AS OCTAL 1211)                                 |
| OCT1400   | OCT | 01400    |                                                        |
| V06N22    | VN  | 0622     |                                                        |

|    |                                |        |            |                                        |           |  |  |  |
|----|--------------------------------|--------|------------|----------------------------------------|-----------|--|--|--|
| 1  | # FIXED + FIXED CONSTANT + USE |        |            |                                        | PAGE 1094 |  |  |  |
| 2  | # MID5                         | OCT    | 1740       | # PINBALL                              |           |  |  |  |
| 3  | BITS2-10                       | OCT    | 1776       |                                        |           |  |  |  |
| 4  | LOW10                          | OCT    | 1777       |                                        |           |  |  |  |
| 5  | # BIT11                        | OCT    | 02000      |                                        |           |  |  |  |
| 6  | # 2K+3                         | OCT    | 2003       | # PINBALL                              |           |  |  |  |
| 7  | LOW7+2K                        | OCT    | 2177       | # OP CODE MASK + BANK 1 FBANK SETTING. |           |  |  |  |
| 8  | EBANK5                         | OCT    | 02400      |                                        |           |  |  |  |
| 9  | PRI03                          | OCT    | 03000      |                                        |           |  |  |  |
| 10 | EBANK7                         | OCT    | 03400      |                                        |           |  |  |  |
| 11 | # LOW11                        | OCT    | 3777       | # PINBALL                              |           |  |  |  |
| 12 | # BIT12                        | OCT    | 04000      |                                        |           |  |  |  |
| 13 | # RELTAB                       | OCT    | 04025      | # T4RUPT                               |           |  |  |  |
| 14 | PRI05                          | OCT    | 05000      |                                        |           |  |  |  |
| 15 | PRI06                          | OCT    | 06000      |                                        |           |  |  |  |
| 16 | PRI07                          | OCT    | 07000      |                                        |           |  |  |  |
| 17 |                                |        |            |                                        |           |  |  |  |
| 18 | # BIT13                        | OCT    | 10000      |                                        |           |  |  |  |
| 19 | #                              | OCT    | 10003      | # T4RUPT RELTAB +1D                    |           |  |  |  |
| 20 | # 13,7,2                       | OCT    | 10102      | # P20-P25 SUNDANCE                     |           |  |  |  |
| 21 | PRI011                         | OCT    | 11000      |                                        |           |  |  |  |
| 22 | # PRI012                       | OCT    | 12000      | # BANKCALL                             |           |  |  |  |
| 23 | PRI013                         | OCT    | 13000      |                                        |           |  |  |  |
| 24 | PRI014                         | OCT    | 14000      |                                        |           |  |  |  |
| 25 | #                              | OCT    | 14031      | # T4RUPT RELTAB +2D                    |           |  |  |  |
| 26 | PRI015                         | OCT    | 15000      |                                        |           |  |  |  |
| 27 | PRI016                         | OCT    | 16000      |                                        |           |  |  |  |
| 28 | # 85DEGS                       | DEC    | .45556     | # P20-P25 SUNDANCE (OCTAL 16450)       |           |  |  |  |
| 29 | PRI017                         | OCT    | 17000      |                                        |           |  |  |  |
| 30 | OCT17770                       | OCT    | 17770      |                                        |           |  |  |  |
| 31 | # BIT14                        | OCT    | 20000      |                                        |           |  |  |  |
| 32 | #                              | OCT    | 20033      | # T4RUPT RELTAB +3D                    |           |  |  |  |
| 33 | PRI021                         | OCT    | 21000      |                                        |           |  |  |  |
| 34 |                                | BLOCK  | 03         |                                        |           |  |  |  |
| 35 |                                | COUNT* | \$\$/FCONS |                                        |           |  |  |  |
| 36 | PRI022                         | OCT    | 22000      | # SERVICE ROUTINES                     |           |  |  |  |
| 37 | PRI023                         | OCT    | 23000      |                                        |           |  |  |  |
| 38 | PRI024                         | OCT    | 24000      |                                        |           |  |  |  |
| 39 | # 5/8+1                        | OCT    | 24001      | # SINGLE PRECISION SUBROUTINES         |           |  |  |  |
| 40 | #                              | OCT    | 24017      | # T4RUPT RELTAB +4D                    |           |  |  |  |
| 41 | PRI025                         | OCT    | 25000      |                                        |           |  |  |  |
| 42 | PRI026                         | OCT    | 26000      |                                        |           |  |  |  |
| 43 | PRI027                         | OCT    | 27000      |                                        |           |  |  |  |
| 44 | # CHRPRIO                      | OCT    | 30000      | # PINBALL                              |           |  |  |  |
| 45 | #                              | OCT    | 30036      | # T4RUPT RELTAB +5D                    |           |  |  |  |
| 46 | PRI031                         | OCT    | 31000      |                                        |           |  |  |  |
| 47 | C1/2                           | DEC    | .7853134   | # (OCTAL 31103)                        |           |  |  |  |
| 48 | PRI032                         | OCT    | 32000      |                                        |           |  |  |  |
| 49 | PRI033                         | OCT    | 33000      |                                        |           |  |  |  |
| 50 | PRI034                         | OCT    | 34000      |                                        |           |  |  |  |
| 51 | #                              | OCT    | 34034      | # T4RUPT RELTAB +6D                    |           |  |  |  |
| 52 |                                |        |            |                                        |           |  |  |  |
| 53 |                                |        |            |                                        |           |  |  |  |
| 54 |                                |        |            |                                        |           |  |  |  |
| 55 |                                |        |            |                                        |           |  |  |  |
| 56 |                                |        |            |                                        |           |  |  |  |
| 57 |                                |        |            |                                        |           |  |  |  |
| 58 |                                |        |            |                                        |           |  |  |  |
| 59 |                                |        |            |                                        |           |  |  |  |
| 60 |                                |        |            |                                        |           |  |  |  |

|    |                               |     |           |                                   |    |
|----|-------------------------------|-----|-----------|-----------------------------------|----|
| 1  | # FIXED-TAXED-CORRECTANT-1002 |     |           |                                   | 1  |
| 2  | PRI035                        | OCT | 35000     |                                   | 2  |
| 3  | PRI036                        | OCT | 36000     |                                   | 3  |
| 4  | PRI037                        | OCT | 37000     |                                   | 4  |
| 5  | 63/64+1                       | OCT | 37401     |                                   | 5  |
| 6  | # MID7                        | OCT | 37600     | # PINBALL                         | 6  |
| 7  | OCT37766                      | OCT | 37766     |                                   | 7  |
| 8  | OCT37774                      | OCT | 37774     |                                   | 8  |
| 9  | OCT37776                      | OCT | 37776     |                                   | 9  |
| 10 | # DPOSMAX                     | OCT | 37777     |                                   | 10 |
| 11 | # BIT15                       | OCT | 40000     |                                   | 11 |
| 12 | # OCT40001                    | OCT | 40001     | # INTERPRETER ( CS 1 INSTRUCTION) | 12 |
| 13 | DLOADCOD                      | OCT | 40014     |                                   | 13 |
| 14 | DLOAD*                        | OCT | 40015     |                                   | 14 |
| 15 | #                             | OCT | 40023     | # T4RUPT RELTAB +7D               | 15 |
| 16 | BIT15+6                       | OCT | 40040     |                                   | 16 |
| 17 | OCT40200                      | OCT | 40200     |                                   | 17 |
| 18 | #                             | OCT | 44035     | # T4RUPT RELTAB +8D               | 18 |
| 19 | #                             | OCT | 50037     | # T4RUPT RELTAB +9D               | 19 |
| 20 | #                             | OCT | 54000     | # T4RUPT RELTAB +10D              | 20 |
| 21 | -BIT14                        | OCT | 57777     |                                   | 21 |
| 22 | # RELTAB11                    | OCT | 60000     | # T4RUPT                          | 22 |
| 23 | C3/2                          | DEC | -.3216147 | # (OCTAL 65552)                   | 23 |
| 24 | 13,14,15                      | OCT | 70000     |                                   | 24 |
| 25 | -1/8                          | OCT | 73777     |                                   | 25 |
| 26 | HIGH4                         | OCT | 74000     |                                   | 26 |
| 27 | -ENDERAS                      | DEC | -2001     | # (OCTAL 74056)                   | 27 |
| 28 | # HI5                         | OCT | 76000     | # PINBALL                         | 28 |
| 29 | HIGH9                         | OCT | 77700     |                                   | 29 |
| 30 | # -ENDVAC                     | DEC | -45       | # INTERPRETER (OCTAL 77722)       | 30 |
| 31 | # -OCT10                      | OCT | -10       | # (OCT 77767)                     | 31 |
| 32 | # NEG4                        | DEC | -4        | # (OCTAL 77773)                   | 32 |
| 33 | NEG3                          | DEC | -3        |                                   | 33 |
| 34 | NEG2                          | OCT | 77775     |                                   | 34 |
| 35 | NEGONE                        | DEC | -1        |                                   | 35 |
| 36 |                               |     |           |                                   | 36 |
| 37 |                               |     |           |                                   | 37 |
| 38 |                               |     |           |                                   | 38 |
| 39 |                               |     |           |                                   | 39 |
| 40 |                               |     |           |                                   | 40 |
| 41 |                               |     |           |                                   | 41 |
| 42 |                               |     |           |                                   | 42 |
| 43 |                               |     |           |                                   | 43 |
| 44 |                               |     |           |                                   | 44 |
| 45 |                               |     |           |                                   | 45 |
| 46 |                               |     |           |                                   | 46 |
| 47 |                               |     |           |                                   | 47 |
| 48 |                               |     |           |                                   | 48 |
| 49 |                               |     |           |                                   | 49 |
| 50 |                               |     |           |                                   | 50 |
| 51 |                               |     |           |                                   | 51 |
| 52 |                               |     |           |                                   | 52 |
| 53 |                               |     |           |                                   | 53 |
| 54 |                               |     |           |                                   | 54 |
| 55 |                               |     |           |                                   | 55 |
| 56 |                               |     |           |                                   | 56 |
| 57 |                               |     |           |                                   | 57 |
| 58 |                               |     |           |                                   | 58 |
| 59 |                               |     |           |                                   | 59 |
| 60 |                               |     |           |                                   | 60 |

## # DEFINED BY EQUALS

# IT WOULD BE TO THE USERS ADVANTAGE TO OCCASIONALLY CHECK ANY OF THESE SYMBOLS IN ORDER TO PREVENT ANY  
# ACCIDENTAL DEFINITION CHANGES.

MINUS1 = NEG1  
NEG1 = NEGONE

ONE = BIT1  
TWO = BIT2  
THREE = OCTAL3

LOW2 = THREE  
FOUR = BIT3  
SIX = REVCNT

LOW3 = SEVEN  
EIGHT = BIT4  
NINE = R2D1

TEN = BINCON  
NOUTCON = ELEVEN  
OCT23 = VD1

OCT25 = MD1  
PRIO1 = BIT10  
EBANK3 = OCT1400

PRIO2 = BIT11  
OCT120 = SUPER101  
OCT140 = SUPER110

2K = BIT11  
EBANK4 = BIT11  
PRIO4 = BIT12

EBANK6 = PRIO3  
QUARTER = BIT13  
PRIO10 = BIT13

OCT10001 = CCSL  
POS1/2 = HALF  
PRIO20 = BIT14

HALF = BIT14  
PRIO30 = CHRPRIO  
BIT13-14 = PRIO30 # INTERPRETER USES IN PROCESSING STORECODE

OCT30002 = TLOAD +1  
B12T14 = PRIO34  
NEGMAX = BIT15

VLOADCOD = BIT15  
VLOAD\* = OCT40001  
OCT60000 = RELTAB11

BANKMASK = HI5

|    |          |        |            |    |
|----|----------|--------|------------|----|
| 1  |          |        |            | 1  |
| 2  |          | SETLOC | INTPRET1   | 2  |
| 3  |          | BANK   |            | 3  |
| 4  |          |        |            | 4  |
| 5  |          | COUNT* | \$\$/ICONS | 5  |
| 6  | DP1/4TH  | 2DEC   | .25        | 6  |
| 7  |          |        |            | 7  |
| 8  | UNITZ    | 2DEC   | 0          | 8  |
| 9  |          |        |            | 9  |
| 10 | UNITY    | 2DEC   | 0          | 10 |
| 11 |          |        |            | 11 |
| 12 | UNITX    | 2DEC   | .5         | 12 |
| 13 |          |        |            | 13 |
| 14 | ZEROVECS | 2DEC   | 0          | 14 |
| 15 |          |        |            | 15 |
| 16 |          | 2DEC   | 0          | 16 |
| 17 |          |        |            | 17 |
| 18 |          | 2DEC   | 0          | 18 |
| 19 |          |        |            | 19 |
| 20 | DPHALF   | =      | UNITX      | 20 |
| 21 | DPPOSMAX | OCT    | 37777      | 21 |
| 22 |          | OCT    | 37777      | 22 |
| 23 |          |        |            | 23 |
| 24 |          |        |            | 24 |
| 25 |          |        |            | 25 |
| 26 |          |        |            | 26 |
| 27 |          |        |            | 27 |
| 28 |          |        |            | 28 |
| 29 |          |        |            | 29 |
| 30 |          |        |            | 30 |
| 31 |          |        |            | 31 |
| 32 |          |        |            | 32 |
| 33 |          |        |            | 33 |
| 34 |          |        |            | 34 |
| 35 |          |        |            | 35 |
| 36 |          |        |            | 36 |
| 37 |          |        |            | 37 |
| 38 |          |        |            | 38 |
| 39 |          |        |            | 39 |
| 40 |          |        |            | 40 |
| 41 |          |        |            | 41 |
| 42 |          |        |            | 42 |
| 43 |          |        |            | 43 |
| 44 |          |        |            | 44 |
| 45 |          |        |            | 45 |
| 46 |          |        |            | 46 |
| 47 |          |        |            | 47 |
| 48 |          |        |            | 48 |
| 49 |          |        |            | 49 |
| 50 |          |        |            | 50 |
| 51 |          |        |            | 51 |
| 52 |          |        |            | 52 |
| 53 |          |        |            | 53 |
| 54 |          |        |            | 54 |
| 55 |          |        |            | 55 |
| 56 |          |        |            | 56 |
| 57 |          |        |            | 57 |
| 58 |          |        |            | 58 |
| 59 |          |        |            | 59 |
| 60 |          |        |            | 60 |

# INTERPRETIVE CONSTANTS IN THE OTHER HALF-MEMORY

SETLOC INTPRET2  
BANK

ZUNIT COUNT\* \$\$/ICONS  
2DEC 0

YUNIT 2DEC 0

XUNIT 2DEC .5

ZEROVEC 2DEC 0

2DEC 0

2DEC 0

DFC-6 OCT 77777 # -0, -6, -12 MUST REMAIN IN THIS ORDER  
DEC -6

DFC-12 DEC -12  
LODPMAX 2OCT 3777737777 # THESE TWO CONSTANTS MUST REMAIN

LODPMAX1 2OCT 3777737777 # ADJACENT AND THE SAME FOR INTEGRATION

ZERODP = ZEROVEC  
HALFDP = XUNIT

|    |                                    |        |            |    |
|----|------------------------------------|--------|------------|----|
| 1  |                                    |        |            | 1  |
| 2  | BLOCK 02                           |        |            | 2  |
| 3  |                                    |        |            | 3  |
| 4  | # SINGLE PRECISION SINE AND COSINE |        |            | 4  |
| 5  |                                    |        |            | 5  |
| 6  |                                    | COUNT* | \$\$/INTER | 6  |
| 7  | SPCOS                              | AD     | HALF       | 7  |
| 8  | SPSIN                              | TS     | TEMK       | 8  |
| 9  | # ARGUMENTS SCALED AT PI           |        |            | 9  |
| 10 |                                    |        |            | 10 |
| 11 | SPT                                | TCF    | SPT        | 11 |
| 12 |                                    |        |            | 12 |
| 13 |                                    |        |            | 13 |
| 14 |                                    |        |            | 14 |
| 15 |                                    |        |            | 15 |
| 16 |                                    |        |            | 16 |
| 17 |                                    |        |            | 17 |
| 18 |                                    |        |            | 18 |
| 19 |                                    |        |            | 19 |
| 20 |                                    |        |            | 20 |
| 21 |                                    |        |            | 21 |
| 22 |                                    |        |            | 22 |
| 23 |                                    |        |            | 23 |
| 24 |                                    |        |            | 24 |
| 25 |                                    |        |            | 25 |
| 26 |                                    |        |            | 26 |
| 27 |                                    |        |            | 27 |
| 28 |                                    |        |            | 28 |
| 29 |                                    |        |            | 29 |
| 30 |                                    |        |            | 30 |
| 31 |                                    |        |            | 31 |
| 32 |                                    |        |            | 32 |
| 33 |                                    |        |            | 33 |
| 34 |                                    |        |            | 34 |
| 35 |                                    |        |            | 35 |
| 36 |                                    |        |            | 36 |
| 37 |                                    |        |            | 37 |
| 38 |                                    |        |            | 38 |
| 39 |                                    |        |            | 39 |
| 40 |                                    |        |            | 40 |
| 41 |                                    |        |            | 41 |
| 42 |                                    |        |            | 42 |
| 43 |                                    |        |            | 43 |
| 44 |                                    |        |            | 44 |
| 45 |                                    |        |            | 45 |
| 46 |                                    |        |            | 46 |
| 47 |                                    |        |            | 47 |
| 48 |                                    |        |            | 48 |
| 49 |                                    |        |            | 49 |
| 50 |                                    |        |            | 50 |
| 51 |                                    |        |            | 51 |
| 52 |                                    |        |            | 52 |
| 53 |                                    |        |            | 53 |
| 54 |                                    |        |            | 54 |
| 55 |                                    |        |            | 55 |
| 56 |                                    |        |            | 56 |
| 57 |                                    |        |            | 57 |
| 58 |                                    |        |            | 58 |
| 59 |                                    |        |            | 59 |
| 60 |                                    |        |            | 60 |

BLOCK 02

# TO ENTER A JOB REQUEST REQUIRING NO VAC AREA:

|       |        |           |                                         |
|-------|--------|-----------|-----------------------------------------|
| NOVAC | COUNT* | \$\$/EXEC |                                         |
|       | INHINT |           |                                         |
|       | AD     | FAKEPRET  | # LOC(MPAC +6) - LOC(QPRET)             |
|       | TS     | NEWPRIO   | # PRIORITY OF NEW JOB + NOVAC C(FIXLOC) |

|        |   |  |                                     |
|--------|---|--|-------------------------------------|
| EXTEND |   |  |                                     |
| INDEX  | Q |  | # Q WILL BE UNDISTURBED THROUGHOUT. |
| DCA    | 0 |  | # 2CADR OF JOB ENTERED.             |

|      |          |
|------|----------|
| DXCH | NEWLOC   |
| CAF  | EXECBANK |

|     |          |                         |
|-----|----------|-------------------------|
| XCH | FBANK    |                         |
| TS  | EXECTEM1 |                         |
| TCF | NOVAC2   | # ENTER EXECUTIVE BANK. |

# TO ENTER A JOB REQUEST REQUIREING A VAC AREA -- E.G., ALL (PARTIALLY) INTERPRETIVE JOBS.

|         |        |         |
|---------|--------|---------|
| FINDVAC | INHINT |         |
|         | TS     | NEWPRIO |
|         | EXTEND |         |

|         |       |          |                                   |
|---------|-------|----------|-----------------------------------|
| SPVACIN | INDEX | Q        |                                   |
|         | DCA   | 0        |                                   |
|         | DXCH  | NEWLOC   |                                   |
|         | CAF   | EXECBANK |                                   |
|         | XCH   | FBANK    |                                   |
|         | TCF   | FINDVAC2 | # OFF TO EXECUTIVE SWITCHED-BANK. |

# TO ENTER A FINDVAC WITH THE PRIORITY IN NEWPRIO TO THE 2CADR ARRIVING IN A AND L:  
# USERS OF SPVAC MUST INHINT BEFORE STORING IN NEWPRIO.

|       |     |         |
|-------|-----|---------|
| SPVAC | XCH | Q       |
|       | AD  | NEG2    |
|       | XCH | Q       |
|       | TCF | SPVACIN |

# TO SUSPEND A BASIC JOB SO A HIGHER PRIORITY JOB MAY BE SERVICED:

|        |      |          |
|--------|------|----------|
| CHANG1 | LXCH | Q        |
|        | CAF  | EXECBANK |
|        | XCH  | BBANK    |
|        | TCF  | CHANJOB  |

# TO SUSPEND AN INTERPRETIVE JOB:

|                                 |    |     |                                          |
|---------------------------------|----|-----|------------------------------------------|
| CHANG2                          | CS | LOC | # NEGATIVE LOC SHOWS JOB = INTERPRETIVE. |
| # ITRACE (4) REFERS TO "CHANG2" | TS | L   |                                          |





# EXECUTIVE

PAGE 1104



|    |    |     |            |    |
|----|----|-----|------------|----|
| 1  |    |     |            | 1  |
| 2  | +2 | CAF | EXECBANK   | 2  |
| 3  |    | TS  | BBANK      | 3  |
| 4  |    | TCF | CHANJOB -1 | 4  |
| 5  |    |     |            | 5  |
| 6  |    |     |            | 6  |
| 7  |    |     |            | 7  |
| 8  |    |     |            | 8  |
| 9  |    |     |            | 9  |
| 10 |    |     |            | 10 |
| 11 |    |     |            | 11 |
| 12 |    |     |            | 12 |
| 13 |    |     |            | 13 |
| 14 |    |     |            | 14 |
| 15 |    |     |            | 15 |
| 16 |    |     |            | 16 |
| 17 |    |     |            | 17 |
| 18 |    |     |            | 18 |
| 19 |    |     |            | 19 |
| 20 |    |     |            | 20 |
| 21 |    |     |            | 21 |
| 22 |    |     |            | 22 |
| 23 |    |     |            | 23 |
| 24 |    |     |            | 24 |
| 25 |    |     |            | 25 |
| 26 |    |     |            | 26 |
| 27 |    |     |            | 27 |
| 28 |    |     |            | 28 |
| 29 |    |     |            | 29 |
| 30 |    |     |            | 30 |
| 31 |    |     |            | 31 |
| 32 |    |     |            | 32 |
| 33 |    |     |            | 33 |
| 34 |    |     |            | 34 |
| 35 |    |     |            | 35 |
| 36 |    |     |            | 36 |
| 37 |    |     |            | 37 |
| 38 |    |     |            | 38 |
| 39 |    |     |            | 39 |
| 40 |    |     |            | 40 |
| 41 |    |     |            | 41 |
| 42 |    |     |            | 42 |
| 43 |    |     |            | 43 |
| 44 |    |     |            | 44 |
| 45 |    |     |            | 45 |
| 46 |    |     |            | 46 |
| 47 |    |     |            | 47 |
| 48 |    |     |            | 48 |
| 49 |    |     |            | 49 |
| 50 |    |     |            | 50 |
| 51 |    |     |            | 51 |
| 52 |    |     |            | 52 |
| 53 |    |     |            | 53 |
| 54 |    |     |            | 54 |
| 55 |    |     |            | 55 |
| 56 |    |     |            | 56 |
| 57 |    |     |            | 57 |
| 58 |    |     |            | 58 |
| 59 |    |     |            | 59 |
| 60 |    |     |            | 60 |

1412THE

# EXECUTIVE

# TO VOLUNTARILY SUSPEND A JOB UNTIL THE COMPLETION OF SOME ANTICIPATED EVENT (I/O EVENT ETC.):

|           |     |          |
|-----------|-----|----------|
| JOB_SLEEP | TS  | LOC      |
|           | CAF | EXECBANK |
|           | TS  | FBANK    |
|           | TCF | JOB_SLP1 |

# TO AWAKEN A JOB PUT TO SLEEP IN THE ABOVE FASHION:

|          |        |           |                                         |
|----------|--------|-----------|-----------------------------------------|
| JOB_WAKE | INHINT |           |                                         |
|          | TS     | NEWLOC    |                                         |
|          | CS     | TWO       | # EXIT IS VIA FINDVAC/NOVAC PROCEDURES. |
|          | ADS    | Q         |                                         |
|          | CAF    | EXECBANK  |                                         |
|          | XCH    | FBANK     |                                         |
|          | TCF    | JOB_WAKE2 |                                         |

# TO CHANGE THE PRIORITY OF A JOB CURRENTLY UNDER EXECUTION:

|          |        |          |                                         |
|----------|--------|----------|-----------------------------------------|
| PRIOCHNG | INHINT |          | # NEW PRIORITY ARRIVES IN A. RETURNS TO |
|          | TS     | NEWPRIO  | # CALLER AS SOON AS NEW JOB PRIORITY IS |
|          | CAF    | EXECBANK | # HIGHEST. PREPARE FOR POSSIBLE BASIC-  |
|          | XCH    | BBANK    | # STYLE CHANGE-JOB.                     |
|          | TS     | BANKSET  |                                         |
|          | CA     | Q        |                                         |
|          | TCF    | PRIOCH2  |                                         |

# TO REMOVE A JOB FROM EXECUTIVE CONSIDERATIONS:

|          |       |           |                                    |
|----------|-------|-----------|------------------------------------|
| ENDOFJOB | CAF   | EXECBANK  |                                    |
|          | TS    | FBANK     |                                    |
|          | TCF   | ENDJOB1   |                                    |
| ENDFIND  | CA    | EXECTEM1  | # RETURN TO CALLER AFTER JOB ENTRY |
|          | TS    | FBANK     | # COMPLETE.                        |
| EXECBANK | TCF   | Q+2       |                                    |
|          | CADR  | FINDVAC2  |                                    |
| FAKEPRET | ADRES | MPAC -36D | # LOC(MPAC +6) - LOC(QPRET)        |

# LOCATE AN AVAILABLE VAC AREA

|          |        |           |                                            |
|----------|--------|-----------|--------------------------------------------|
|          | BANK   | 01        |                                            |
|          | COUNT* | \$\$/EXEC |                                            |
| FINDVAC2 | TS     | EXECTEM1  | # (SAVE CALLER'S BANK FIRST.)              |
|          | CCS    | VAC1USE   |                                            |
|          | TCF    | VACFOUND  |                                            |
|          | CCS    | VAC2USE   |                                            |
|          | TCF    | VACFOUND  |                                            |
|          | CCS    | VAC3USE   |                                            |
|          | TCF    | VACFOUND  |                                            |
|          | CCS    | VAC4USE   |                                            |
|          | TCF    | VACFOUND  |                                            |
|          | CCS    | VAC5USE   |                                            |
|          | TCF    | VACFOUND  |                                            |
|          | LXCH   | EXECTEM1  |                                            |
|          | CA     | Q         |                                            |
|          | TC     | BAILOUT1  |                                            |
|          | OCT    | 1201      | # NO VAC AREAS.                            |
| VACFOUND | AD     | TWO       | # RESERVE THIS VAC AREA BY STORING A ZERO  |
|          | ZL     |           | # IN ITS VAC USE REGISTER AND STORE THE    |
|          | INDEX  | A         | # ADDRESS OF THE FIRST WORD OF IT IN THE   |
|          | LXCH   | 0         | # LOW NINE BITS OF THE PRIORITY WORD.      |
|          | ADS    | NEWPRIO   |                                            |
| NOVAC2   | CAF    | ZERO      | # NOVAC ENTERS HERE. FIND A CORE SET.      |
|          | TS     | LOCCTR    |                                            |
|          | CAF    | NO.CORES  | # SEVEN SETS OF ELEVEN REGISTERS EACH.     |
| NOVAC3   | TS     | EXECTEM2  |                                            |
|          | INDEX  | LOCCTR    |                                            |
|          | CCS    | PRIORITY  | # EACH PRIORITY REGISTER CONTAINS -0 IF    |
|          | TCF    | NEXTCORE  | # THE CORESPONDING CORE SET IS AVAILABLE.  |
| NO.CORES | DEC    | 7         |                                            |
|          | TCF    | NEXTCORE  | # AN ACTIVE JOB HAS A POSITIVE PRIORITY    |
|          |        |           | # BUT A DORMANT JOB'S PRIORITY IS NEGATIVE |

CORFOUND

CA  
INDEXNEWPRIO  
LOCCTR# SET THE PRIORITY OF THIS JOB IN THE CORE  
# SET'S PRIORITY REGISTER AND SET THETS  
MASK  
INDEXPRIORITY  
LOW9  
LOCCTR# JOB'S PUSH-DOWN POINTER AT THE BEGINNING  
# OF THE WORK AREA AND OVERFLOW INDICATOR.

TS

PUSHLOC

# OFF TO PREPARE FOR INTERPRETIVE PROGRAMS.

CCS

LOCCTR

# IF CORE SET ZERO IS BEING LOADED, SET UP  
# OVFINDD AND FIXLOC IMMEDIATELY.

TCF

SETLOC

TS

OVFINDD

CA

PUSHLOC

TS

FIXLOC

SPECTEST

CCS

NEWJOB

# SEE IF ANY ACTIVE JOBS WAITING (RARE).

TCF

SETLOC

# MUST BE AWAKENED OUT UNCHANGED JOB.

TC

CCSHOLE

TC

CCSHOLE

TS

NEWJOB

# +0 SHOWS ACTIVE JOB ALREADY SET.

DXCH

NEWLOC

DXCH

LOC

TCF

ENDFIND

SETLOC

DXCH

NEWLOC

# SET UP THE LOCATION REGISTERS FOR THIS

INDEX

LOCCTR

DXCH

LOC

INDEX

NEWJOB

# THIS INDEX INSTRUCTION INSURES THAT THE  
# HIGHEST ACTIVE PRIORITY WILL BE COMPARED  
# WITH THE NEW PRIORITY TO SEE IF NEWJOB  
# SHOULD BE SET TO SIGNAL A SWITCH.

CS

PRIORITY

AD

NEWPRIO

EXTEND

BZMF

ENDFIND

CA

LOCCTR

TS

NEWJOB

# LOCCTR IS LEFT SET AT THIS CORE SET IF  
# THE CALLER WANTS TO LOAD ANY MPAC  
# REGISTERS, ETC.

TCF

ENDFIND

NEXTCORE

CAF

COREINC

ADS

LOCCTR

CCS

EXECTEM2

TCF

NOVAC3

LXCH

EXECTEM1

CA

Q

TC

BAILOUT1

# NO CORE SETS AVAILABLE.

OCT

1202

# THE FOLLOWING ROUTINE SWAPS CORE SET 0 WITH THAT WHOSE RELATIVE ADDRESS IS IN NEWJOB.

|         |        |         |                                           |
|---------|--------|---------|-------------------------------------------|
| -2      | LXCH   | LOC     |                                           |
| -1      | CAE    | BANKSET | # BANKSET, NOT BBANK, HAS RIGHT CONTENTS. |
| CHANJOB | INHINT |         |                                           |

|  |        |          |                                   |
|--|--------|----------|-----------------------------------|
|  | EXTEND |          |                                   |
|  | ROR    | SUPERBNK | # PICK UP CURRENT SBANK FOR BBCON |
|  | XCH    | L        | # LOC IN A AND BBCON IN L.        |

|    |       |        |                         |
|----|-------|--------|-------------------------|
| +4 | INDEX | NEWJOB | # SWAP LOC AND BANKSET. |
|    | DXCH  | LOC    |                         |
|    | DXCH  | LOC    |                         |

|  |        |         |  |
|--|--------|---------|--|
|  | CAE    | BANKSET |  |
|  | EXTEND |         |  |

|  |       |          |                                         |
|--|-------|----------|-----------------------------------------|
|  | WRITE | SUPERBNK | # SET SBANK FOR NEW JOB.                |
|  | DXCH  | MPAC     | # SWAP MULTI-PURPOSE ACCUMULATOR AREAS. |
|  | INDEX | NEWJOB   |                                         |

|  |      |      |    |
|--|------|------|----|
|  | DXCH | MPAC |    |
|  | DXCH | MPAC |    |
|  | DXCH | MPAC | +2 |

|  |       |        |    |
|--|-------|--------|----|
|  | INDEX | NEWJOB |    |
|  | DXCH  | MPAC   | +2 |
|  | DXCH  | MPAC   | +2 |

|  |       |        |    |
|--|-------|--------|----|
|  | DXCH  | MPAC   | +4 |
|  | INDEX | NEWJOB |    |
|  | DXCH  | MPAC   | +4 |

|  |       |        |    |
|--|-------|--------|----|
|  | DXCH  | MPAC   | +4 |
|  | DXCH  | MPAC   | +6 |
|  | INDEX | NEWJOB |    |

|  |      |      |    |
|--|------|------|----|
|  | DXCH | MPAC | +6 |
|  | DXCH | MPAC | +6 |

|  |        |        |                                  |
|--|--------|--------|----------------------------------|
|  | CAF    | ZERO   |                                  |
|  | XCH    | OVFIND | # MAKE PUSHLOC NEGATIVE IF OVFIN |
|  | EXTEND |        |                                  |

|  |     |         |  |
|--|-----|---------|--|
|  | BZF | +3      |  |
|  | CS  | PUSHLOC |  |
|  | TS  | PUSHLOC |  |

|  |       |         |  |
|--|-------|---------|--|
|  | DXCH  | PUSHLOC |  |
|  | INDEX | NEWJOB  |  |

|  |      |         |                                   |
|--|------|---------|-----------------------------------|
|  | DXCH | PUSHLOC |                                   |
|  | DXCH | PUSHLOC | # SWAPS PUSHLOC AND PRIORITY.     |
|  | CAF  | LOW9    | # SET FIXLOC TO BASE OF VAC AREA. |

|  |      |          |  |
|--|------|----------|--|
|  | MASK | PRIORITY |  |
|  | TS   | FIXLOC   |  |

|  |     |          |                                       |
|--|-----|----------|---------------------------------------|
|  | CCS | PUSHLOC  | # SET OVERFLOW INDICATOR ACCORDING TO |
|  | CAF | ZERO     |                                       |
|  | TCF | ENDPRCHG | -1                                    |



|    |          |        |         |    |
|----|----------|--------|---------|----|
| 1  |          |        |         | 1  |
| 2  |          | CS     | PUSHLOC | 2  |
| 3  |          | TS     | PUSHLOC | 3  |
| 4  |          | CAF    | ONE     | 4  |
| 5  |          | XCH    | OVFIND  | 5  |
| 6  |          | TS     | NEWJOB  | 6  |
| 7  |          |        |         | 7  |
| 8  | ENDPRCHG | RELINT |         | 8  |
| 9  |          | DXCH   | LOC     | 9  |
| 10 |          | EXTEND |         | 10 |
| 11 |          | BZMF   | +2      | 11 |
| 12 |          | DTCB   |         | 12 |
| 13 |          |        |         | 13 |
| 14 |          |        |         | 14 |
| 15 |          |        |         | 15 |
| 16 |          |        |         | 16 |
| 17 |          |        |         | 17 |
| 18 |          |        |         | 18 |
| 19 |          |        |         | 19 |
| 20 |          |        |         | 20 |
| 21 |          |        |         | 21 |
| 22 |          |        |         | 22 |
| 23 |          |        |         | 23 |
| 24 |          |        |         | 24 |
| 25 |          |        |         | 25 |
| 26 |          |        |         | 26 |
| 27 |          |        |         | 27 |
| 28 |          |        |         | 28 |
| 29 |          |        |         | 29 |
| 30 |          |        |         | 30 |
| 31 |          |        |         | 31 |
| 32 |          |        |         | 32 |
| 33 |          |        |         | 33 |
| 34 |          |        |         | 34 |
| 35 |          |        |         | 35 |
| 36 |          |        |         | 36 |
| 37 |          |        |         | 37 |
| 38 |          |        |         | 38 |
| 39 |          |        |         | 39 |
| 40 |          |        |         | 40 |
| 41 |          |        |         | 41 |
| 42 |          |        |         | 42 |
| 43 |          |        |         | 43 |
| 44 |          |        |         | 44 |
| 45 |          |        |         | 45 |
| 46 |          |        |         | 46 |
| 47 |          |        |         | 47 |
| 48 |          |        |         | 48 |
| 49 |          |        |         | 49 |
| 50 |          |        |         | 50 |
| 51 |          |        |         | 51 |
| 52 |          |        |         | 52 |
| 53 |          |        |         | 53 |
| 54 |          |        |         | 54 |
| 55 |          |        |         | 55 |
| 56 |          |        |         | 56 |
| 57 |          |        |         | 57 |
| 58 |          |        |         | 58 |
| 59 |          |        |         | 59 |
| 60 |          |        |         | 60 |

# BASIC JOBS HAVE POSITIVE ADDRESSES, SO  
# DISPATCH WITH A DTCB.  
# IF INTERPRETIVE, SET UP EBANK, ETC.

|    |                                   |        |            |                                           |    |
|----|-----------------------------------|--------|------------|-------------------------------------------|----|
| 1  |                                   |        |            |                                           | 1  |
| 2  |                                   | COM    |            | # EPILOGUE TO JOB CHANGE FOR INTERPRETIVE | 2  |
| 3  |                                   | AD     | ONE        |                                           | 3  |
| 4  |                                   | TS     | LOC        | # RESUME                                  | 4  |
| 5  |                                   | TCF    | INTRSM     |                                           | 5  |
| 6  |                                   |        |            |                                           | 6  |
| 7  | # COMPLETE JOBSLEEP PREPARATIONS. |        |            |                                           | 7  |
| 8  |                                   |        |            |                                           | 8  |
| 9  | JOBSLP1                           | INHINT |            |                                           | 9  |
| 10 |                                   | CS     | PRIORITY   | # NNZ PRIORITY SHOWS JOB ASLEEP.          | 10 |
| 11 |                                   | TS     | PRIORITY   |                                           | 11 |
| 12 |                                   | CAF    | LOW7       |                                           | 12 |
| 13 |                                   | MASK   | BBANK      |                                           | 13 |
| 14 |                                   | EXTEND |            |                                           | 14 |
| 15 |                                   | ROR    | SUPERBNK   | # SAVE OLD SUPERBANK VALUE.               | 15 |
| 16 |                                   | TS     | BANKSET    |                                           | 16 |
| 17 |                                   | CS     | ZERO       |                                           | 17 |
| 18 | JOBSLP2                           | TS     | BUF +1     | # HOLDS -- HIGHEST PRIORITY.              | 18 |
| 19 |                                   | TCF    | EJSCAN     | # SCAN FOR HIGHEST PRIORITY ALA ENDOFJOB. | 19 |
| 20 |                                   |        |            |                                           | 20 |
| 21 | NUCHANG2                          | INHINT |            | # QUICK... DON'T LET NEWJOB CHANGE TO +0. | 21 |
| 22 |                                   | CCS    | NEWJOB     |                                           | 22 |
| 23 |                                   | TCF    | +3         | # NEWJOB STILL PNZ                        | 23 |
| 24 |                                   | RELINT |            | # NEW JOB HAS CHANGED TO +0. WAKE UP JOB  | 24 |
| 25 |                                   | TCF    | ADVAN +2   | # VIA NUDIRECT. (VERY RARE CASE.)         | 25 |
| 26 |                                   |        |            |                                           | 26 |
| 27 |                                   | CAF    | TWO        |                                           | 27 |
| 28 |                                   | EXTEND |            |                                           | 28 |
| 29 |                                   | WOR    | DSALMOUT   | # TURN ON ACTIVITY LIGHT                  | 29 |
| 30 |                                   | DXCH   | LOC        | # AND SAVE ADDRESS INFO FOR BENEFIT OF    | 30 |
| 31 |                                   | TCF    | CHANJOB +4 | # POSSIBLE SLEEPINT JOB.                  | 31 |
| 32 |                                   |        |            |                                           | 32 |
| 33 |                                   |        |            |                                           | 33 |
| 34 |                                   |        |            |                                           | 34 |
| 35 |                                   |        |            |                                           | 35 |
| 36 |                                   |        |            |                                           | 36 |
| 37 |                                   |        |            |                                           | 37 |
| 38 |                                   |        |            |                                           | 38 |
| 39 |                                   |        |            |                                           | 39 |
| 40 |                                   |        |            |                                           | 40 |
| 41 |                                   |        |            |                                           | 41 |
| 42 |                                   |        |            |                                           | 42 |
| 43 |                                   |        |            |                                           | 43 |
| 44 |                                   |        |            |                                           | 44 |
| 45 |                                   |        |            |                                           | 45 |
| 46 |                                   |        |            |                                           | 46 |
| 47 |                                   |        |            |                                           | 47 |
| 48 |                                   |        |            |                                           | 48 |
| 49 |                                   |        |            |                                           | 49 |
| 50 |                                   |        |            |                                           | 50 |
| 51 |                                   |        |            |                                           | 51 |
| 52 |                                   |        |            |                                           | 52 |
| 53 |                                   |        |            |                                           | 53 |
| 54 |                                   |        |            |                                           | 54 |
| 55 |                                   |        |            |                                           | 55 |
| 56 |                                   |        |            |                                           | 56 |
| 57 |                                   |        |            |                                           | 57 |
| 58 |                                   |        |            |                                           | 58 |
| 59 |                                   |        |            |                                           | 59 |
| 60 |                                   |        |            |                                           | 60 |

# TO WAKE UP A JOB, EACH CORE SET IS FOUND TO LOCATE ALL JOBS WHICH ARE ASLEEP. IF THE FCADR IN THE  
# LOC REGISTER OF ANY SUCH JOB MATCHES THAT SUPPLIED BY THE CALLER, THAT JOB IS AWAKENED. IF NO JOB IS FOUND,  
# LOCCTR IS SET TO -1 AND NO FURTHER ACTION TAKES PLACE.

JOBWAKE2 TS EXECTEM1  
CAF ZERO # BEGIN CORE SET SCAN  
TS LOCCTR

JOBWAKE4 TS EXECTEM2  
INDEX LOCCTR  
CCS PRIORITY

COREINC TCF JOBWAKE3 # ACTIVE JOB -- CHECK NEXT CORE SET.  
DEC 12 # 12 REGISTERS PER CORE SET.  
TCF WAKETEST # SLEEPING JOB -- SEE IF CADR MATCHES.

JOBWAKE3 CAF COREINC  
ADS LOCCTR  
CCS EXECTEM2  
TCF JOBWAKE4  
CS ONE # EXIT IF SLEEPIN JOB NOT FOUND.  
TS LOCCTR  
TCF ENDFIND

WAKETEST CS NEWLOC  
INDEX LOCCTR  
AD LOC  
EXTEND  
BZF +2 # IF MATCH.  
TCF JOBWAKE3 # EXAMINE NEXT CORE SET IF NO MATCH.

INDEX LOCCTR # RE-COMPLEMENT PRIORITY TO SHOW JOB AWAKE  
CS PRIORITY  
TS NEWPRIO  
INDEX LOCCTR  
TS PRIORITY

CS FBANKMSK # MAKE UP THE 2CADR OF THE WAKE ADDRESS  
MASK NEWLOC # USING THE CADR IN NEWLOC AND THE EBANK  
AD 2K # HALF OF BBANK SAVED IN BANKSET.

XCH NEWLOC  
MASK FBANKMSK  
INDEX LOCCTR  
AD BANKSET  
TS NEWLOC +1

CCS LOCCTR # SPECIAL TREATMENT IF THIS JOB WAS  
TCF SETLOC # ALREADY IN THE RUN (0) POSITION.  
TCF SPECTEST





# EXECUTIVE

PAGE 1112

1412THE

|    |                                                                                              |      |          |                                           |    |
|----|----------------------------------------------------------------------------------------------|------|----------|-------------------------------------------|----|
| 1  | # PRIORITY CHANGE. CHANGE THE CONTENTS OF PRIORITY AND SCAN FOR THE JOB OF HIGHEST PRIORITY. |      |          |                                           | 1  |
| 2  |                                                                                              |      |          |                                           | 2  |
| 3  |                                                                                              |      |          |                                           | 3  |
| 4  | PRIOCH2                                                                                      | TS   | LOC      |                                           | 4  |
| 5  |                                                                                              | CAF  | ZERO     | # SET FLAG TO TELL ENDJOB SCANNER IF THIS | 5  |
| 6  |                                                                                              | TS   | BUF      | # JOB IS STILL HIGHEST PRIORITY.          | 6  |
| 7  |                                                                                              | CAF  | LOW9     |                                           | 7  |
| 8  |                                                                                              | MASK | PRIORITY |                                           | 8  |
| 9  |                                                                                              | AD   | NEWPRIO  |                                           | 9  |
| 10 |                                                                                              | TS   | PRIORITY |                                           | 10 |
| 11 |                                                                                              | COM  |          |                                           | 11 |
| 12 |                                                                                              | TCF  | JOBSLP2  | # AND TO EJSCAN.                          | 12 |
| 13 |                                                                                              |      |          |                                           | 13 |
| 14 |                                                                                              |      |          |                                           | 14 |
| 15 |                                                                                              |      |          |                                           | 15 |
| 16 |                                                                                              |      |          |                                           | 16 |
| 17 |                                                                                              |      |          |                                           | 17 |
| 18 |                                                                                              |      |          |                                           | 18 |
| 19 |                                                                                              |      |          |                                           | 19 |
| 20 |                                                                                              |      |          |                                           | 20 |
| 21 |                                                                                              |      |          |                                           | 21 |
| 22 |                                                                                              |      |          |                                           | 22 |
| 23 |                                                                                              |      |          |                                           | 23 |
| 24 |                                                                                              |      |          |                                           | 24 |
| 25 |                                                                                              |      |          |                                           | 25 |
| 26 |                                                                                              |      |          |                                           | 26 |
| 27 |                                                                                              |      |          |                                           | 27 |
| 28 |                                                                                              |      |          |                                           | 28 |
| 29 |                                                                                              |      |          |                                           | 29 |
| 30 |                                                                                              |      |          |                                           | 30 |
| 31 |                                                                                              |      |          |                                           | 31 |
| 32 |                                                                                              |      |          |                                           | 32 |
| 33 |                                                                                              |      |          |                                           | 33 |
| 34 |                                                                                              |      |          |                                           | 34 |
| 35 |                                                                                              |      |          |                                           | 35 |
| 36 |                                                                                              |      |          |                                           | 36 |
| 37 |                                                                                              |      |          |                                           | 37 |
| 38 |                                                                                              |      |          |                                           | 38 |
| 39 |                                                                                              |      |          |                                           | 39 |
| 40 |                                                                                              |      |          |                                           | 40 |
| 41 |                                                                                              |      |          |                                           | 41 |
| 42 |                                                                                              |      |          |                                           | 42 |
| 43 |                                                                                              |      |          |                                           | 43 |
| 44 |                                                                                              |      |          |                                           | 44 |
| 45 |                                                                                              |      |          |                                           | 45 |
| 46 |                                                                                              |      |          |                                           | 46 |
| 47 |                                                                                              |      |          |                                           | 47 |
| 48 |                                                                                              |      |          |                                           | 48 |
| 49 |                                                                                              |      |          |                                           | 49 |
| 50 |                                                                                              |      |          |                                           | 50 |
| 51 |                                                                                              |      |          |                                           | 51 |
| 52 |                                                                                              |      |          |                                           | 52 |
| 53 |                                                                                              |      |          |                                           | 53 |
| 54 |                                                                                              |      |          |                                           | 54 |
| 55 |                                                                                              |      |          |                                           | 55 |
| 56 |                                                                                              |      |          |                                           | 56 |
| 57 |                                                                                              |      |          |                                           | 57 |
| 58 |                                                                                              |      |          |                                           | 58 |
| 59 |                                                                                              |      |          |                                           | 59 |
| 60 |                                                                                              |      |          |                                           | 60 |

# RELEASE THIS CORE SET AND VAC AREA AND SCAN FOR THE JOB OF HIGHEST ACTIVE PRIORITY.

ENDJOB1

INHINT

CS ZERO

TS BUF +1

XCH PRIORITY

MASK LOW9

TS L

CS FAKEPRET

AD L

EXTEND

BZMF EJSCAN # NOVAC ENDOFJOB

CCS L

INDEX A

TS 0

EJSCAN

CCS PRIORITY +12D

TC EJ1

TC CCSHOLE

TCF +1

CCS PRIORITY +24D # EXAMINE EACH PRIORITY REGISTER TO FIND

TC EJ1 # THE JOB OF HIGHEST ACTIVE PRIORITY.

TC CCSHOLE

TCF +1

-CCSPR

CCS PRIORITY +36D

TC EJ1

-CCS PRIORITY

TCF +1

CCS PRIORITY +48D

TC EJ1

TC CCSHOLE

TCF +1

CCS PRIORITY +60D

TC EJ1

TC CCSHOLE

TCF +1

CCS PRIORITY +72D

TC EJ1

TC CCSHOLE

TCF +1

CCS PRIORITY +84D

|     |         |
|-----|---------|
| TC  | EJ1     |
| TC  | CCSHOLE |
| TCF | +1      |

# EXECUTIVE

# EVALUATE THE RESULTS OF THE SCAN.

CCS BUF +1 # SEE IF THERE ARE ANY ACTIVE JOBS WAITING  
TC CCSHOLE  
TC CCSHOLE

TCF +2  
TCF DUMMYJOB

CCS BUF # BUF IS ZERO IS THIS IS A PRIOCHNG AND  
TCF +2 # CHANGED PRIORITY IS STILL HIGHEST.  
TCF ENDPRCHG -1

INDEX A # OTHERWISE, SET NEWJOB TO THE RELATIVE  
CAF 0 -1 # ADDRESS OF THE NEW JOB'S CORE SET.

AD -CCSPR  
TS NEWJOB  
TCF CHANJOB -2

EJ1 TS BUF +2  
AD BUF +1 # - OLD HIGH PRIORITY.

CCS A  
CS BUF +2  
TCF EJ2 # NEW HIGH PRIORITY.

NOOP  
INDEX Q  
TC 2 # PROCEED WITH SEARCH.

EJ2 TS BUF +1  
EXTEND

QXCH BUF # FOR LOCATING CCS PRIORITY + X INSTR.  
INDEX BUF  
TC 2

# IDLING AND COMPUTER ACTIVITY (GREEN) LIGHT MAINTENANCE. THE IDLING ROUTINE IS NOT A JOB IN ITSELF,  
# BUT RATHER A SUBROUTINE OF THE EXECUTIVE.

EBANK= SELFRET # SELF-CHECK STORAGE IN EBANK.

DUMMYJOB CS ZERO # SET NEWJOB TO -0 FOR IDLING.  
TS NEWJOB  
RELINT

CS TWO # TURN OFF THE ACTIVITY LIGHT.  
EXTEND  
WAND DSALMOUT

ADVAN CCS NEWJOB # IS THE NEWJOB ACTIVE?  
TCF NUCHANG2 # YES... ONE REQUIRING A CHANGE JOB.  
CAF TWO # NEW JOB ALREADY IN POSITION FOR  
TCF NUDIRECT # EXECUTION

CA SELFRET  
TS L # PUT RETURN ADDRESS IN L.  
CAF SELF BANK  
TCF SUPDXCHZ +1 # AND DISPATCH JOB.

SELF BANK EBANK= SELFRET  
BBCON SELFCHK

NUDIRECT EXTEND # TURN THE GREEN LIGHT BACK ON.  
WOR DSALMOUT

DXCH LOC # JOBS STARTED IN THIS FASHION MUST BE  
TCF SUPDXCHZ

BLOCK 2 # IN FIXED-FIXED SO OTHERS MAY USE.

COUNT\* \$\$/EXEC

# SUPDXCHZ -- ROUTINE TO TRANSFER TO SUPEBANK.  
# CALLING SEQUENCE:

# TCF SUPDXCHZ # WITH 2CADR OF DESIRED LOCATION IN A + L.

SUPDXCHZ XCH L # BASIC.

+1 EXTEND  
WRITE SUPERBNK  
TS BBANK

TC L

NEG100 OCT 77677

## # PROGRAM DESCRIPTION

DATE -- 10 OCTOBER 1966

# MOD NO -- 2

LOG SECTION -- WAITLIST

# MOD BY -- MILLER (DTMAX INCREASED TO 162.5 SEC)

ASSEMBLY -- SUNBURST REV 5

# MOD 3 BY KERNAN (INHINT INSERTED AT WAITLIST) 2/28/68 SKIPPER REV 4

# MOD 4 BY KERNAN (TWIDDLE IN 54) 3/28/68 SKIPPER REV 13.

## # FUNCTIONAL DESCRIPTION --

# PART OF A SECTION OF PROGRAMS -- WAITLIST, TASKOVER, T3RUPT, USED TO CALL A PROGRAM (CALLED A TASK),  
# WHICH IS TO BEGIN IN C(A) CENTISECONDS. WAITLIST UPDATES TIME3, LST1, AND LST2. THE MEANING OF THESE LISTS  
# FOLLOW.

# C(TIME3) = 16384 -(T1-T) CENTISECONDS, (T=PRESENT TIME, T1=TIME FOR TASK1)

# C(LST1) = -(T2-T1)+1

# C(LST1 +1) = -(T3-T2)+1

# C(LST1 +2) = -(T4-T3)+1

# ...

# C(LST1 +6) = -(T8-T7)+1

# C(LST1 +7) = -(T9-T8)+1

# C(LST2) = 2CADR OF TASK1

# C(LST2 +2) = 2CADR OF TASK2

# ...

# C(LST2 +14) = 2CADR OF TASK8

# C(LST2 +16) = 2CADR OF TASK9

## # WARNINGS --

# 1) 1 <= C(A) <= 16250D (1 CENTISECOND TO 162.5 SEC)

# 2) 9 TASKS MAXIMUM

# 3) TASKS CALLED UNDER INTERRUPT INHIBITED

# 4) TASKS END BY TC TASKOVER

## # CALLING SEQUENCE --

# L-1 CA DELTAT (TIME IN CENTISECONDS TO TASK START)

# L TC WAITLIST

# L+1 2CADR DESIRED TASK.

# L+2 (MINOR OF 2CADR)

# L+3 RELINT (RETURNS HERE)

## # TWIDDLE --

# TWIDDLE IS FOR USE WHEN THE TASK BEING SET UP IS IN THE SAME EBANK AND FBANK AS THE USER. IN  
# SUCH CASES, IT IMPROVES UPON WAITLIST BY ELIMINATING THE NEED FOR THE BBCON HALF OF THE 2CADR,

```
1 # SAVING A WORD. TWIDDLE IS LIKE WAITLIST IN EVERY RESPECT EXCEPT CALLING SEQUENCE, TO WIT,
2 #
3 # L-1 CA DELTAT
4 # L TC TWIDDLE
5 # L+1 ADRES DESIRED TASK
6 # L+2 RELINT (RETURNS HERE)
7 #
8 # NORMAL EXIT MODES --
9 # AT L+3 OF CALLING SEQUENCE.
10 #
11 # ALARM OR ABORT EXIT MODES --
12 # TC ABORT
13 # OCT 1203 (WAITLIST OVERFLOW -- TOO MANY TASKS)
14 #
15 # ERASABLE INITIALIZATION REQUIRED --
16 # ACCOMPLISHED BY FRESH START -- LST2, ..., LST2 +16 = ENDTASK
17 # LST1, ..., LST1 +7 = NEG1/2
18 #
19 # OUTPUT --
20 # LST1 AND LST2 UPDATED WITH NEW TASK AND ASSOCIATED TIME.
21 #
22 # DEBRIS --
23 # CENTRALS -- A,Q,L
24 # OTHER -- WAITEXIT, WAITADR, WAITTEMP, WAITBANK
25 #
26 # DETAILED ANALYSIS OF TIMING --
27 # CONTROL WILL NOT BE RETURNED TO THE SPECIFIED ADDRESS (2CADR) IN EXACTLY DELTA T CENTISECONDS.
28 # THE APPROXIMATE TIME MAY BE CALCULATED AS FOLLOWS:
29 # LET TO = THE TIME OF THE TC WAITLIST
30 # LET TS = TO +147U + COUNTER INCREMENTS (SET UP TIME)
31 # LET X = TS -(100TS)/100 (VARIANCE FROM COUNTERS)
32 # LET Y = LENGTH OF TIME OF INHIBIT INTERRUPT AFTER T3RUPT
33 # LET Z = LENGTH OF TIME TO PROCESS TASKS WHICH ARE DUE THIS T3RUPT BUT DISPATCHED EARLIER.
34 # (Z=0, USUALLY).
35 # LET DELTD = THE ACTUAL TIME TAKEN TO GIVE CONTROL TO 2CADR
36 # THEN DELTD = TS+DELTA T -X +Y +Z +1.05MS* +COUNTERS*
37 # *THE TIME TAKEN BY WAITLIST ITSELF AND THE COUNTER TICKING DURING THIS WAITLIST TIME.
38 # IN SHORT, THE ACTUAL TIME TO RETURN CONTROL TO A 2CADR IS AUGMENTED BY THE TIME TO SET UP THE TASK'S
39 # INTERRUPT, ALL COUNTERS TICKING, THE T3RUPT PROCESSING TIME, THE WAITLIST PROCESSING TIME AND THE POSSIBILITY
40 # OF OTHER TASKS INHIBITING THE INTERRUPT.
```

BLOCK 02

```
1
2 EBANK= LST1 # TASK LISTS IN SWITCHED E BANK.
3
4 COUNT* $$/WAIT
5 TWIDDLE INHINT
6 TS L # SAVE DELAY TIME IN L
7
8 CA POSMAX
9 ADS Q # CREATING OVERFLOW AND Q-1 IN Q
10 CA BBANK
11
12 EXTEND
13 ROR SUPERBNK
14 XCH L
15
16 WAITLIST INHINT
17 XCH Q # SAVE DELTA T IN Q AND RETURN IN
18 TS WAITEXIT # WAITEXIT.
19 EXTEND
20 INDEX WAITEXIT # IF TWIDDLING, THE TS SKIPS TO HERE
21
22 -1 DCA 0 # PICK UP 2CADR OF TASK.
23 DLY2 TS WAITADR # BBCON WILL REMAIN IN L
24 CAF WAITBB # ENTRY FROM FIXDELAY AND VARDELAY.
25 XCH BBANK
26 TCF WAIT2
27
28 # RETURN TO CALLER AFTER TASK INSERTION:
29
30 LVWTLIST DXCH WAITEXIT
31 AD TWO
32 DTCB
33
34 WAITBB EBANK= LST1
35 BBCON WAIT2
36
37 # RETURN TO CALLER +2 AFTER WAITING DT SPECIFIED AT CALLER +1.
38
39 FIXDELAY INDEX Q # BOTH ROUTINES MUST BE CALLED UNDER
40 CAF 0 # WAITLIST CONTROL AND TERMINATE THE TASK
41 INCR Q # IN WHICH THEY WERE CALLED.
42
43 # RETURN TO CALLER +1 AFTER WAITING THE DT AS ARRIVING IN A.
44
45 VARDELAY XCH Q # DT TO Q. TASK ADRES TO WAITADR.
46 TS WAITADR # BBANK IS SAVED DURING DELAY.
47 CA BBANK
48 EXTEND
49 ROR SUPERBNK # ADD SBANK TO BBCON.
50 TS L
51 CAF DELAYEX
52 TS WAITEXIT # GO TO TASKOVER AFTER TASK ENTRY.
53 TCF DLY2
54
55
56
57
58
59
60
```





# WAITLIST

PAGE 1120



1412THE

1  
2 DELAYEX TCF TASKOVER -2 # RETURNS TO TASKOVER.  
3  
4

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80



# WAITLIST

PAGE 1121

# ENDTASK MUST ENTERED IN FIXED-FIXED SO IT IS DISTINGUISHABLE BY ITS ADRES ALONE.

ENDTASK           EBANK=   LST1  
                  -2CADR   SVCT3

SVCT3           CCS       FLAGWRD2       # DRIFT FLAG  
                 TCF       TASKOVER  
                 TCF       TASKOVER  
                 TCF       +1

CKIMUSE          CCS       IMUCADR       # DON'T DO NBDONLY IF SOMEONE ELSE IS IN  
                 TCF       SVCT3X       # IMUSTALL.  
                 TCF       +3  
                 TCF       SVCT3X  
                 TCF       SVCT3X

+3               CAF       PRI035       # COMPENSATE FOR NBD COEFFICIENTS ONLY.  
                 TC       NOVAC       #       ENABLE EVERY 81.93 SECONDS  
                 EBANK=   NBDX  
                 2CADR   NBDONLY

TCF       TASKOVER

SVCT3X          TC       FIXDELAY       # DELAY MAX OF 2 TIMES FOR IMUZERO.  
                 DEC       500  
                 TC       SVCT3       # CHECK DRIFT FLAG AGAIN.

1412THE

# BEGIN TASK INSERTION.

WAIT2            BANK    01  
                 COUNT\* \$\$/WAIT  
                 TS     WAITBANK        # BBANK OF CALLING PROGRAM.

                 CA     Q  
                 EXTEND  
                 BZMF    WAITPOOH

                 CS     TIME3  
                 AD     BIT8

# BIT 8 = OCT 200

CCS    A

# TEST 200 - C(TIME3). IF POSITIVE,  
# IT MEANS THAT TIME3 OVERFLOW HAS OCCURRED PRIOR TO CS TIME3 AND THAT  
# C(TIME3) = T - T1, INSTEAD OF 1.0 - (T1 - T). THE FOLLOWING FOUR  
# ORDERS SET C(A) = TD - T1 + 1 IN EITHER CASE.

                 AD     OCT40001  
                 CS     A

# OVERFLOW HAS OCCURRED. SET C(A) =  
# T - T1 + 1.0 - 201

# NORMAL CASE (C(A) NNZ) YIELDS SAME C(A):  $-( -(1.0 - (T1 - T)) + 200) - 1$ 

                 AD     OCT40201  
                 AD     Q

# RESULT = TD - T1 + 1.

CCS    A

# TEST TD - T1 + 1.

                 AD     LST1  
                 TCF    WTLST5

# IF TD - T1 POS, GO TO WTLST5 WITH  
# C(A) = (TD - T1) + C(LST1) = TD - T2 + 1

                 NOOP  
                 CS     Q

# NOTE THAT THIS PROGRAM SECTION IS NEVER ENTERED WHEN T-T1 G/E -1,  
# SINCE TD-T1+1 = (TD-T) + (T-T1+1), AND DELTA T = TD-T G/E +1. (G/E  
# SYMBOL MEANS GREATER THAN OR EQUAL TO). THUS THERE NEED BE NO CON-  
# CERN OVER A PREVIOUS OR IMMINENT OVEFLOW OF TIME3 HERE.

                 AD     POS1/2  
                 AD     POS1/2  
                 XCH    TIME3  
                 AD     NEGMAX

# WHEN TD IS NEXT, FORM QUANTITY

# 1.0 - DELTA T = 1.0 - (TD - T)

                 AD     Q  
                 EXTEND  
                 QXCH    7

# 1.0 - DELTAT T NOW COMPLETE.  
# ZERO INDEX Q.  
# (ZQ)





# WAITLIST

PAGE 1124

|    |        |     |        |                    |    |
|----|--------|-----|--------|--------------------|----|
| 1  | WTLST5 | CCS | A      | # TEST TD - T2 + 1 | 1  |
| 2  |        | AD  | LST1   | +1                 | 2  |
| 3  |        | TCF | +4     |                    | 3  |
| 4  |        | AD  | ONE    |                    | 4  |
| 5  |        | TC  | WTLST2 |                    | 5  |
| 6  |        | OCT | 1      |                    | 6  |
| 7  |        |     |        |                    | 7  |
| 8  |        |     |        |                    | 8  |
| 9  | +4     | CCS | A      | # TEST TD - T3 + 1 | 9  |
| 10 |        | AD  | LST1   | +2                 | 10 |
| 11 |        | TCF | +4     |                    | 11 |
| 12 |        | AD  | ONE    |                    | 12 |
| 13 |        | TC  | WTLST2 |                    | 13 |
| 14 |        | OCT | 2      |                    | 14 |
| 15 |        |     |        |                    | 15 |
| 16 | +4     | CCS | A      | # TEST TD - T4 + 1 | 16 |
| 17 |        | AD  | LST1   | +3                 | 17 |
| 18 |        | TCF | +4     |                    | 18 |
| 19 |        | AD  | ONE    |                    | 19 |
| 20 |        | TC  | WTLST2 |                    | 20 |
| 21 |        | OCT | 3      |                    | 21 |
| 22 |        |     |        |                    | 22 |
| 23 | +4     | CCS | A      | # TEST TD - T5 + 1 | 23 |
| 24 |        | AD  | LST1   | +4                 | 24 |
| 25 |        | TCF | +4     |                    | 25 |
| 26 |        | AD  | ONE    |                    | 26 |
| 27 |        | TC  | WTLST2 |                    | 27 |
| 28 |        | OCT | 4      |                    | 28 |
| 29 |        |     |        |                    | 29 |
| 30 | +4     | CCS | A      | # TEST TD - T6 + 1 | 30 |
| 31 |        | AD  | LST1   | +5                 | 31 |
| 32 |        | TCF | +4     |                    | 32 |
| 33 |        | AD  | ONE    |                    | 33 |
| 34 |        | TC  | WTLST2 |                    | 34 |
| 35 |        | OCT | 5      |                    | 35 |
| 36 |        |     |        |                    | 36 |
| 37 | +4     | CCS | A      | # TEST TD - T7 + 1 | 37 |
| 38 |        | AD  | LST1   | +6                 | 38 |
| 39 |        | TCF | +4     |                    | 39 |
| 40 |        | AD  | ONE    |                    | 40 |
| 41 |        | TC  | WTLST2 |                    | 41 |
| 42 |        | OCT | 6      |                    | 42 |
| 43 |        |     |        |                    | 43 |
| 44 |        |     |        |                    | 44 |
| 45 |        |     |        |                    | 45 |
| 46 |        |     |        |                    | 46 |
| 47 |        |     |        |                    | 47 |
| 48 |        |     |        |                    | 48 |
| 49 |        |     |        |                    | 49 |
| 50 |        |     |        |                    | 50 |
| 51 |        |     |        |                    | 51 |
| 52 |        |     |        |                    | 52 |
| 53 |        |     |        |                    | 53 |
| 54 |        |     |        |                    | 54 |
| 55 |        |     |        |                    | 55 |
| 56 |        |     |        |                    | 56 |
| 57 |        |     |        |                    | 57 |
| 58 |        |     |        |                    | 58 |
| 59 |        |     |        |                    | 59 |
| 60 |        |     |        |                    | 60 |

1412THE



## # WAITLIST

FILLED

|      |
|------|
| DXCH |
| TC   |
| OCT  |

```
WAITEXIT
BAILOUT1
01203
```

## # NO ROOM IN THE INN

# THE ENTRY TC WTLST2 JUST PRECEDING OCT N IS FOR T LE TD LE T -1.  
#  
# (LE MEANS LESS THAN OR EQUAL TO). AT ENTRY, C(A) = -(TD - T + 1)  
#  
# THE LST1 ENTRY -(T -T +1) IS TO BE REPLACED BY -(TD - T + 1), AND  
#  
# THE ENTRY -(T - TD + 1) IS TO BE INSERTED IMMEDIATELY FOLLOWING.  
#

|        |       |          |                          |
|--------|-------|----------|--------------------------|
| WTLST2 | TS    | WAITTEMP | # C(A) = -(TD - T + 1)   |
|        | INDEX | Q        |                          |
|        | CAF   | 0        |                          |
|        | TS    | Q        | # INDEX VALUE INTO Q.    |
|        | CAF   | ONE      |                          |
|        | AD    | WAITTEMP |                          |
|        | INDEX | Q        | # C(A) = -(TD - T ) + 1. |
|        | ADS   | LST1 -1  | #                        |
|        | CS    | WAITTEMP |                          |
|        | INDEX | Q        |                          |
|        | TCF   | WTLST4   |                          |

|   |            |   |                 |
|---|------------|---|-----------------|
| # | C(TIME3)   | = | 1.0 - (T1 - T)  |
| # |            |   |                 |
| # | C(LST1)    | = | - (T2 - T1) + 1 |
| # | C(LST1+1)  | = | - (T3 - T2) + 1 |
| # | C(LST1+2)  | = | - (T4 - T3) + 1 |
| # | C(LST1+3)  | = | - (T5 - T4) + 1 |
| # | C(LST1+4)  | = | - (T6 - T5) + 1 |
| # |            |   |                 |
| # | C(LST2)    | = | 2CADR TASK1     |
| # | C(LST2+2)  | = | 2CADR TASK2     |
| # | C(LST2+4)  | = | 2CADR TASK3     |
| # | C(LST2+6)  | = | 2CADR TASK4     |
| # | C(LST2+8)  | = | 2CADR TASK5     |
| # | C(LST2+10) | = | 2CADR TASK6     |



# ENTERS HERE ON T3 RUPT TO DISPATCH WAITLISTED TASK.

|        |        |          |   |                                  |
|--------|--------|----------|---|----------------------------------|
| T3RUPT | EXTEND |          |   |                                  |
|        | ROR    | SUPERBNK | # | READ CURRENT SUPERBANK VALUE AND |
|        | TS     | BANKRUPT | # | SAVE WITH E AND F BANK VALUES.   |

|        |       |
|--------|-------|
| EXTEND |       |
| QXCH   | QRUPT |

|         |     |        |   |                         |
|---------|-----|--------|---|-------------------------|
| T3RUPT2 | CAF | NEG1/2 | # | DISPATCH WAITLIST TASK. |
|---------|-----|--------|---|-------------------------|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +7 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +6 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +5 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +4 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +3 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +2 |
|-----|------|----|

|     |      |    |
|-----|------|----|
| XCH | LST1 | +1 |
|-----|------|----|

|     |      |  |
|-----|------|--|
| XCH | LST1 |  |
|-----|------|--|

|    |        |  |
|----|--------|--|
| AD | POSMAX |  |
|----|--------|--|

|     |       |  |
|-----|-------|--|
| ADS | TIME3 |  |
|-----|-------|--|

|    |         |  |
|----|---------|--|
| TS | RUPTAGN |  |
|----|---------|--|

|    |      |  |
|----|------|--|
| CS | ZERO |  |
|----|------|--|

|    |         |  |
|----|---------|--|
| TS | RUPTAGN |  |
|----|---------|--|

|        |  |  |   |                |
|--------|--|--|---|----------------|
| EXTEND |  |  | # | DISPATCH TASK. |
|--------|--|--|---|----------------|

|     |         |  |
|-----|---------|--|
| DCS | ENDTASK |  |
|-----|---------|--|

|      |      |      |
|------|------|------|
| DXCH | LST2 | +16D |
|------|------|------|

|      |      |      |
|------|------|------|
| DXCH | LST2 | +14D |
|------|------|------|

|      |      |      |
|------|------|------|
| DXCH | LST2 | +12D |
|------|------|------|

|      |      |      |
|------|------|------|
| DXCH | LST2 | +10D |
|------|------|------|

|      |      |     |
|------|------|-----|
| DXCH | LST2 | +8D |
|------|------|-----|

|      |      |    |
|------|------|----|
| DXCH | LST2 | +6 |
|------|------|----|

|      |      |    |
|------|------|----|
| DXCH | LST2 | +4 |
|------|------|----|

|      |      |    |
|------|------|----|
| DXCH | LST2 | +2 |
|------|------|----|

|      |      |  |
|------|------|--|
| DXCH | LST2 |  |
|------|------|--|

|     |   |  |
|-----|---|--|
| XCH | L |  |
|-----|---|--|

|        |  |  |
|--------|--|--|
| EXTEND |  |  |
|--------|--|--|

|       |          |   |                                   |
|-------|----------|---|-----------------------------------|
| WRITE | SUPERBNK | # | SET SUPERBANK FROM BBCON OF 2CADR |
|-------|----------|---|-----------------------------------|

|     |   |   |                          |
|-----|---|---|--------------------------|
| XCH | L | # | RESTORE TO L FOR DXCH Z. |
|-----|---|---|--------------------------|

|      |  |  |
|------|--|--|
| DTCB |  |  |
|------|--|--|



|    |                                                |        |           |    |
|----|------------------------------------------------|--------|-----------|----|
| 1  | # RETURN, AFTER EXECUTION OF T3 OVERFLOW TASK: |        |           | 1  |
| 2  |                                                |        |           | 2  |
| 3  |                                                |        |           | 3  |
| 4  | TASKOVER                                       | BLOCK  | 02        | 4  |
| 5  |                                                | COUNT* | \$\$/WAIT | 5  |
| 6  |                                                | CCS    | RUPTAGN   | 6  |
| 7  | # IF +1 RETURN TO T3RUPT, IF -0 RESUME.        |        |           | 7  |
| 8  |                                                | CAF    | WAITBB    | 8  |
| 9  |                                                | TS     | BBANK     | 9  |
| 10 |                                                | TCF    | T3RUPT2   | 10 |
| 11 | # DISPATCH NEXT TASK IF IT WAS DUE.            |        |           | 11 |
| 12 |                                                |        |           | 12 |
| 13 | RESUME                                         | CA     | BANKRUPT  | 13 |
| 14 |                                                | EXTEND |           | 14 |
| 15 |                                                | WRITE  | SUPERBNK  | 15 |
| 16 | # RESTORE SUPERBANK BEFORE RESUME IS DONE      |        |           | 16 |
| 17 | NOQRSM                                         | QXCH   | QRUPT     | 17 |
| 18 |                                                | CA     | BANKRUPT  | 18 |
| 19 |                                                | XCH    | BBANK     | 19 |
| 20 | NOQBRSM                                        | DXCH   | ARUPT     | 20 |
| 21 |                                                | RELINT |           | 21 |
| 22 |                                                | RESUME |           | 22 |
| 23 |                                                |        |           | 23 |
| 24 |                                                |        |           | 24 |
| 25 |                                                |        |           | 25 |
| 26 |                                                |        |           | 26 |
| 27 |                                                |        |           | 27 |
| 28 |                                                |        |           | 28 |
| 29 |                                                |        |           | 29 |
| 30 |                                                |        |           | 30 |
| 31 |                                                |        |           | 31 |
| 32 |                                                |        |           | 32 |
| 33 |                                                |        |           | 33 |
| 34 |                                                |        |           | 34 |
| 35 |                                                |        |           | 35 |
| 36 |                                                |        |           | 36 |
| 37 |                                                |        |           | 37 |
| 38 |                                                |        |           | 38 |
| 39 |                                                |        |           | 39 |
| 40 |                                                |        |           | 40 |
| 41 |                                                |        |           | 41 |
| 42 |                                                |        |           | 42 |
| 43 |                                                |        |           | 43 |
| 44 |                                                |        |           | 44 |
| 45 |                                                |        |           | 45 |
| 46 |                                                |        |           | 46 |
| 47 |                                                |        |           | 47 |
| 48 |                                                |        |           | 48 |
| 49 |                                                |        |           | 49 |
| 50 |                                                |        |           | 50 |
| 51 |                                                |        |           | 51 |
| 52 |                                                |        |           | 52 |
| 53 |                                                |        |           | 53 |
| 54 |                                                |        |           | 54 |
| 55 |                                                |        |           | 55 |
| 56 |                                                |        |           | 56 |
| 57 |                                                |        |           | 57 |
| 58 |                                                |        |           | 58 |
| 59 |                                                |        |           | 59 |
| 60 |                                                |        |           | 60 |

```
1 # LONGCALL
2 # PROGRAM DESCRIPTION DATE -- 17 MARCH 1967
3 # PROGRAM WRITTEN BY W.H.VANDEVER LOG SECTION WAITLIST
4 # MOD BY -- R. MELANSON TO ADD DOCUMENTATION ASSEMBLY SUNDISK REV. 100
5 #
6 # FUNCTIONAL DESCRIPTION --
7 # LONGCALL IS CALLED WITH THE DELTA TIME ARRIVING IN A,L SCALED AS TIME2,TIME1 WITH THE 2CADR OF THE TASK
8 # IMMEDIATELY FOLLOWING THE TC LONGCALL. FOR EXAMPLE, IT MIGHT BE DONE AS FOLLOWS WHERE TIMELOC IS THE NAME OF
9 # A DP REGISTER CONTAINING A DELTA TIME AND WHERE TASKTODO IS THE NAME OF THE LOCATION AT WHICH LONGCALL IS TO
10 # START.
11 # CALLING SEQUENCE --
12 # EXTEND
13 # DCA TIMELOC
14 # TC LONGCALL
15 # 2CADR TASKTODO
16 # NORMAL EXIT MODE --
17 # 1) TC WAITLIST
18 # 2) DTCB (TC L+3 OF CALLING ROUTINE 1ST PASS THRU LONGCYCL)
19 # 3) DTCB (TO TASKOVER ON SUBSEQUENT PASSES THRU LONGCYCL)
20 # ALARM OR ABORT EXIT MODE --
21 # NONE
22 # OUTPUT --
23 # LONGTIME AND LONGTIME+1 = DELTA TIME
24 # LONGEXIT AND LONGEXIT+1 = RETURN 2CADR
25 # LONGCADR AND LONGCADR+1 = TASK 2CADR
26 # A = SINGLE PRECISION TIME FOR WAITLIST
27 # ERASABLE INITIALIZATION --
28 # A = MOST SIGNIFICANT PART OF DELTA TIME
29 # L = LEAST SIGNIFICANT PART OF DELTA TIME
30 # Q = ADDRESS OF 2CADR TASK VALUE
31 # DEBRIS --
32 # A,Q,L
33 # LONGCADR AND LONGCADR+1
34 # LONGEXIT AND LONGEXIT+1
35 # LONGTIME AND LONGTIME+1
36 # *** THE FOLLOWING IS TO BE IN FIXED-FIXED AND UNSWITCHED ERRASIBLE **
37
38 BLOCK 02
39 EBANK= LST1
40 LONGCALL DXCH LONGTIME # OBTAIN THE DELTA TIME
41
42 EXTEND # OBTAIN THE 2CADR
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1
2 NDX Q
3 DCA 0
4 DXCH LONGCADR
5
6 EXTEND # NOW GO TO THE APPROPRIATE SWITCHED BANK
7 DCA LGCL2CDR # FOR THE REST OF LONGCALL
8 DTCB
9
10 LGCL2CDR EBANK= LST1
11 2CADR LNGCALL2
12
13 # *** THE FOLLOWING MAY BE IN A SWITCHED BANK, INCLUDING ITS ERASABLE ***
14
15 BANK 01
16 COUNT* $$/WAIT
17 LXCH LONGEXIT +1 # SAVE THE CORRECT BB FOR RETURN
18 CA TWO # OBTAIN THE RETURN ADDRESS
19
20 ADS Q
21 TS LONGEXIT
22
23 CA LONGTIME # CHECK FOR LEGITIMATE DELTA-TIME
24 CCS A
25 TCF LONGCYCL # HI-ORDER OK --> ALL IS OK.
26 TCF +2 # HI-ORDER ZERO --> CHECK LO-ORDER.
27 TCF LONGPOOH # HI-ORDER NEG. --> NEG. DT
28 CA LONGTIME +1 # CHECK LO-ORDER FOR ZERO OR NEGATIVE.
29
30 EXTEND
31 BZMF LONGPOOH # BAD DELTA-TIME. ABORT
32
33 # *** WAITLIST TASK LONGCYCL ***
34
35 LONGCYCL EXTEND # CAN WE SUCCESFULLY TAKE ABOUT 1.25
36 DCS # MINUTES OFF OF LONGTIME
37 DAS LONGTIME
38
39 CCS LONGTIME +1 # THE REASONING BEHIND THIS PART IS
40 TCF MUCHTIME # INVOLVED, TAKING INTO ACCOUNT THAT THE
41 # WORDS MAY NOT BE SIGNED CORRECTED (DP
42 # BASIC INSTRUCTIONS
43 # DO NOT SIGN CORRECT) AND THAT WE SUBTRAC-
44 # TED BIT14 (1 OVER HALF THE POS. VALUE
45 # REPRESENTABLE IN SINGLE WORD)
46 # CAN'T GET HERE *****
47
48 NOOP
49 TCF +1
50
51 CCS LONGTIME
52 TCF MUCHTIME
53
54 DPBIT14 OCT 00000
55 OCT 20000
56
57 # LONGCALL
```

# WAITLIST

|    |                                 |        |             |                                             |    |
|----|---------------------------------|--------|-------------|---------------------------------------------|----|
| 1  |                                 |        |             |                                             | 1  |
| 2  | LASTTIME                        | CA     | BIT14       | # GET BACK THE CORRECT DELTA T FOR WAITLIST | 2  |
| 3  |                                 | ADS    | LONGTIME +1 |                                             | 3  |
| 4  |                                 | TC     | WAITLIST    |                                             | 4  |
| 5  |                                 | EBANK= | LST1        |                                             | 5  |
| 6  |                                 | 2CADR  | GETCADR     | # THE ENTRY TO OUR LONGCADR                 | 6  |
| 7  |                                 |        |             |                                             | 7  |
| 8  | LONGRTRN                        | CA     | TSKOVCDR    | # SET IT UP SO THAT ONLY THE FIRST EXIT IS  | 8  |
| 9  |                                 | DXCH   | LONGEXIT    | # TO THE CALLER OF LONGCALL                 | 9  |
| 10 |                                 | DTCB   |             | # THE REST ARE TO TASKOVER                  | 10 |
| 11 |                                 |        |             |                                             | 11 |
| 12 | MUCHTIME                        | CA     | BIT14       | # WE HAVE OVER OUR ABOUT 1.25 MINUTES       | 12 |
| 13 |                                 | TC     | WAITLIST    | # SO SET UP FOR ANOTHER CYCLE THROUGH HERE  | 13 |
| 14 |                                 | EBANK= | LST1        |                                             | 14 |
| 15 |                                 | 2CADR  | LONGCYCL    |                                             | 15 |
| 16 |                                 |        |             |                                             | 16 |
| 17 |                                 | TCF    | LONGRTRN    | # NOW EXIT PROPERLY                         | 17 |
| 18 |                                 |        |             |                                             | 18 |
| 19 | # *** WAITLIST TASK GETCADR *** |        |             |                                             | 19 |
| 20 |                                 |        |             |                                             | 20 |
| 21 | GETCADR                         | DXCH   | LONGCADR    | # GET THE LONGCALL THAT WE WISHED TO START  | 21 |
| 22 |                                 | DTCB   |             | # AND TRANSFER CONTROL TO IT                | 22 |
| 23 |                                 |        |             |                                             | 23 |
| 24 | TSKOVCDR                        | GENADR | TASKOVER    |                                             | 24 |
| 25 | LONGPOOH                        | DXCH   | LONGEXIT    |                                             | 25 |
| 26 |                                 | TCF    | +2          |                                             | 26 |
| 27 | WAITPOOH                        | DXCH   | WAITEXIT    |                                             | 27 |
| 28 | +2                              | TC     | POOD001     |                                             | 28 |
| 29 |                                 | OCT    | 01204       |                                             | 29 |
| 30 |                                 |        |             |                                             | 30 |
| 31 |                                 |        |             |                                             | 31 |
| 32 |                                 |        |             |                                             | 32 |
| 33 |                                 |        |             |                                             | 33 |
| 34 |                                 |        |             |                                             | 34 |
| 35 |                                 |        |             |                                             | 35 |
| 36 |                                 |        |             |                                             | 36 |
| 37 |                                 |        |             |                                             | 37 |
| 38 |                                 |        |             |                                             | 38 |
| 39 |                                 |        |             |                                             | 39 |
| 40 |                                 |        |             |                                             | 40 |
| 41 |                                 |        |             |                                             | 41 |
| 42 |                                 |        |             |                                             | 42 |
| 43 |                                 |        |             |                                             | 43 |
| 44 |                                 |        |             |                                             | 44 |
| 45 |                                 |        |             |                                             | 45 |
| 46 |                                 |        |             |                                             | 46 |
| 47 |                                 |        |             |                                             | 47 |
| 48 |                                 |        |             |                                             | 48 |
| 49 |                                 |        |             |                                             | 49 |
| 50 |                                 |        |             |                                             | 50 |
| 51 |                                 |        |             |                                             | 51 |
| 52 |                                 |        |             |                                             | 52 |
| 53 |                                 |        |             |                                             | 53 |
| 54 |                                 |        |             |                                             | 54 |
| 55 |                                 |        |             |                                             | 55 |
| 56 |                                 |        |             |                                             | 56 |
| 57 |                                 |        |             |                                             | 57 |
| 58 |                                 |        |             |                                             | 58 |
| 59 |                                 |        |             |                                             | 59 |
| 60 |                                 |        |             |                                             | 60 |

```
1 # SUBROUTINE TO CONVERT RAD VECTOR AT GIVEN TIME TO LAT, LONG AND ALT
2 #
3 # CALLING SEQUENCE
4 #
5 # L-1 CALL
6 # L LAT-LONG
7 # SUBROUTINES USED
8 #
9 # R-TO-RP, ARCTAN, SETGAMMA, SETRE
10 # ERASABLE INIT. REQ.
11 #
12 # AXO, -AYO, AZO, TEPHEM (SET AT LAUNCH TIME)
13 # ALPHAV = POSITION VECTOR METERS B-29
14 # MPAC-- TIME (CSECS B-28)
15 # ERADFLAG =1, TO COMPUTE EARTH RADIUS, =0 FOR FIXED EARTH RADIUS
16 # LUNAFLAG=0 FOR EARTH, 1 FOR MOON
17 # OUTPUT
18 #
19 # LATITUDE IN LAT (REVS. B-0)
20 # LONGITUDE IN LONG (REVS. B-0)
21 # ALTITUDE IN ALT METERS B-29
22 #
23 # BANK 30
24 # SETLOC LATLONG
25 # BANK
26 #
27 # COUNT* $$/LT-LG
28 # EBANK= ALPHAV
29 # STQ SETPD
30 # INCORPEX
31 #
32 # STOVL 6D # SAVE TIME IN 6-7D FOR R-TO-RP
33 # ALPHAV
34 # PUSH ABVAL # 0-5D= R FOR R-TO-RP
35 # STODL ALPHAM # ABS. VALUE OF R FOR ALT FORMULA BELOW
36 # ZEROVEC # SET MPAC=0 FOR EARTH, NON-ZERO FOR MOON
37 # BOFF COS # USE COS(0) TO GET NON-ZERO IN MPAC
38 # LUNAFLAG # 0=EARTH, 1=MOON
39 # CALLRTRP
40 #
41 # CALLRTRP CALL R-TO-RP # RP VECTOR CONVERTED FROM R B-29
42 # UNIT # UNIT RP B-1
43 # STCALL ALPHAV # U2= 1/2 SINL FOR SETRE SUBR BELOW
44 # SETGAMMA # SET GAMMA=B2/A2 FOR EARTH, =1 FOR MOON
45 # CALL # SCALED B-1
46 # SETRE # CALC RE METERS B-29
47 # DLOAD DSQ
48 # ALPHAV
49 # PDDL DSQ
50 # ALPHAV +2
51 # DAD SQRT
```



|    |  |        |           |                                       |    |
|----|--|--------|-----------|---------------------------------------|----|
| 1  |  |        |           |                                       | 1  |
| 2  |  | DMP    | SL1R      |                                       | 2  |
| 3  |  |        | GAMRP     |                                       | 3  |
| 4  |  | STODL  | COSTH     | # COS(LAT) B-1                        | 4  |
| 5  |  |        | ALPHAV +4 |                                       | 5  |
| 6  |  | STCALL | SINTH     | # SIN(LAT) B-1                        | 6  |
| 7  |  |        | ARCTAN    |                                       | 7  |
| 8  |  | STODL  | LAT       | # LAT B0                              | 8  |
| 9  |  |        | ALPHAV    |                                       | 9  |
| 10 |  | STODL  | COSTH     | # COS(LONG) B-1                       | 10 |
| 11 |  |        | ALPHAV +2 |                                       | 11 |
| 12 |  | STCALL | SINTH     | # SIN(LONG) B-1                       | 12 |
| 13 |  |        | ARCTAN    |                                       | 13 |
| 14 |  | STODL  | LONG      | # LONG. REVS B-0 IN RANGE -1/2 TO 1/2 | 14 |
| 15 |  |        | ALPHAM    |                                       | 15 |
| 16 |  | DSU    |           | # ALT= R-RE METERS B-29               | 16 |
| 17 |  |        | ERADM     |                                       | 17 |
| 18 |  | STCALL | ALT       | # EXIT WITH ALT METERS B-29           | 18 |
| 19 |  |        | INCORPEX  |                                       | 19 |
| 20 |  |        |           |                                       | 20 |
| 21 |  |        |           |                                       | 21 |
| 22 |  |        |           |                                       | 22 |
| 23 |  |        |           |                                       | 23 |
| 24 |  |        |           |                                       | 24 |
| 25 |  |        |           |                                       | 25 |
| 26 |  |        |           |                                       | 26 |
| 27 |  |        |           |                                       | 27 |
| 28 |  |        |           |                                       | 28 |
| 29 |  |        |           |                                       | 29 |
| 30 |  |        |           |                                       | 30 |
| 31 |  |        |           |                                       | 31 |
| 32 |  |        |           |                                       | 32 |
| 33 |  |        |           |                                       | 33 |
| 34 |  |        |           |                                       | 34 |
| 35 |  |        |           |                                       | 35 |
| 36 |  |        |           |                                       | 36 |
| 37 |  |        |           |                                       | 37 |
| 38 |  |        |           |                                       | 38 |
| 39 |  |        |           |                                       | 39 |
| 40 |  |        |           |                                       | 40 |
| 41 |  |        |           |                                       | 41 |
| 42 |  |        |           |                                       | 42 |
| 43 |  |        |           |                                       | 43 |
| 44 |  |        |           |                                       | 44 |
| 45 |  |        |           |                                       | 45 |
| 46 |  |        |           |                                       | 46 |
| 47 |  |        |           |                                       | 47 |
| 48 |  |        |           |                                       | 48 |
| 49 |  |        |           |                                       | 49 |
| 50 |  |        |           |                                       | 50 |
| 51 |  |        |           |                                       | 51 |
| 52 |  |        |           |                                       | 52 |
| 53 |  |        |           |                                       | 53 |
| 54 |  |        |           |                                       | 54 |
| 55 |  |        |           |                                       | 55 |
| 56 |  |        |           |                                       | 56 |
| 57 |  |        |           |                                       | 57 |
| 58 |  |        |           |                                       | 58 |
| 59 |  |        |           |                                       | 59 |
| 60 |  |        |           |                                       | 60 |

```
SUBROUTINE TO CONVERT LAT, LONG, ALT AT GIVEN TIME TO RADIUS VECTOR
CALLING SEQUENCE
```

```
#
L-1 CALL
L LALOTORV
```

```
SUBROUTINES USED
```

```
SETGAMMA, SETRE, RP-TO-R
```

```
ERASABLE INIT. REQ.
```

```
AX0, AYO, AZO, TEPHEM SET AT LAUNCH TIME
```

```
LAT-- LATITUDE (REVS B0)
```

```
LONG-- LONGITUDE (REVS B0)
```

```
ALT-- ALTITUDE (METERS) B-29
```

```
MPAC-- TIME (CSECS B-28)
```

```
ERADFLAG =1 TO COMPUTE EARTH RADIUS, =0 FOR FIXED EARTH RADIUS
```

```
LUNAFLAG=0 FOR EARTH, 1 FOR MOON
```

```
OUTPUT
```

```
#
R-VECTOR IN ALPHAV (METERS B-29)
```

```
LALOTORV STQ SETPD # LAT, LONG, ALT TO R VECTOR
```

```
INCORPEX
OD
```

```
STCALL 6D # 6-7D= TIME FOR RP-TO-R
SETGAMMA # GAMMA=B2/A2 FOR EARTH, 1 FOR MOON B-1
```

```
DLOAD SIN # COS(LONG)COS(LAT) IN MPAC
```

```
DMPR LAT # UNIT RP= SIN(LONG)COS(LAT) 2-3D
```

```
PDDL PDDL # PD 2 GAMMA*SIN(LAT) 0-1D
```

```
GAMRP
```

```
COS LAT # 0-1D= GAMMA*SIN(LAT) B-2
```

```
PDDL PDDL # PD4 2-3D=COS(LAT) B-1 TEMPORARILY
```

```
LONG
```

```
SIN DMPR # PD 2
```

```
PDDL COS # PD 4 2-3D=SIN(LONG)COS(LAT) B-2
```

```
LAT
```

```
PDDL COS # PD 6 4-5D=COS(LAT) B-1 TEMPORARILY
```

```
LONG
```

```
DMPR VDEF # PD 4 MPAC= COS(LONG)COS(LAT) B-2
```

```
UNIT PUSH # 0-5D= UNIT RP FOR RP-TO-R SUBR.
```

```
STCALL ALPHAV # ALPHAV +4= SINL FOR SETRE SUBR.
```

```
SETRE
```

```
DLOAD BOFF # SET MPAC=0 FOR EARTH, NON-ZERO FOR MOON
```

```
ZEROVEC
```

```
LUNAFLAG
```

```
CALLRPRT
```

```
CALLRPRT COS # USE COS(0) TO GET NON-ZERO IN MPAC
```

```
CALL
```

```
STODL RP-TO-R # EXIT WITH UNIT R VECTOR IN MPAC
```

```
ALPHAV
```

```
ERADM
```



DAD VXSC # (RE + ALT)(UNIT R) METERS B-30

ALT

ALPHAV

VSL1

# R METERS B-29

STCALL

ALPHAV

# EXIT WITH R IN METERS B-29

INCORPEX

# SUBROUTINE TO COMPUTE EARTH RADIUS

#

# INPUT

#

# 1/2 SIN LAT IN ALPHAV +4

#

# OUTPUT

#

# EARTH RADIUS IN ERADM AND MPAC (METERS B-29)

GETERAD

DLOAD

DSQ

SL1

ALPHAV +4

# SIN\*\*2(L)

BDSU

DP1/2

# COS\*\*2(L)

DMPR

BDSU

EE

DP1/2

BDDV

SQRT

B2XSC

SR4R

STORE

ERADM

RVQ

# THE FOLLOWING CONSTANTS WERE COMPUTED WITH A=6378166,B=6356784 METERS

# B2XSC= B\*\*2 SCALED B-51

# B2/A2= B\*\*2/A\*\*2 SCALED B-1

# EE=(1-B\*\*2/A\*\*2) SCALED B-0

B2XSC 2DEC .0179450689 # B\*\*2 SCALED B-51

DP1/2 = XUNIT

B2/A2 2DEC .9933064884 B-1 # GAMMA= B\*\*2/A\*\*2 B-1

EE 2DEC 6.6935116 E-3 # (1-B\*\*2/A\*\*2) B-0

|    |                                                            |       |                        |    |
|----|------------------------------------------------------------|-------|------------------------|----|
| 1  | # ARCTAN SUBROUTINE                                        |       |                        | 1  |
| 2  | #                                                          |       |                        | 2  |
| 3  | # CALLING SEQUENCE                                         |       |                        | 3  |
| 4  | #                                                          |       |                        | 4  |
| 5  | # SIN THETA IN SINTH B-1                                   |       |                        | 5  |
| 6  | # COS THETA IN COSTH B-1                                   |       |                        | 6  |
| 7  | # CALL ARCTAN                                              |       |                        | 7  |
| 8  | #                                                          |       |                        | 8  |
| 9  | # OUTPUT                                                   |       |                        | 9  |
| 10 | # ARCTAN THETA IN MPAC AND THETA B-0 IN RANGE -1/2 TO +1/2 |       |                        | 10 |
| 11 |                                                            |       |                        | 11 |
| 12 |                                                            |       |                        | 12 |
| 13 | ARCTAN                                                     | BOV   |                        | 13 |
| 14 |                                                            |       | CLROVFLW               | 14 |
| 15 | CLROVFLW                                                   | DLOAD | DSQ                    | 15 |
| 16 |                                                            |       | SINTH                  | 16 |
| 17 |                                                            | PDDL  | DSQ                    | 17 |
| 18 |                                                            |       | COSTH                  | 18 |
| 19 |                                                            | DAD   |                        | 19 |
| 20 |                                                            | BZE   | SQRT                   | 20 |
| 21 |                                                            |       | ARCTANXX               | 21 |
| 22 |                                                            |       | # ATAN=0/0 SET THETA=0 | 22 |
| 23 |                                                            | BDDV  | BOV                    | 23 |
| 24 |                                                            |       | SINTH                  | 24 |
| 25 |                                                            |       | ATAN=90                | 25 |
| 26 |                                                            | SR1   | ASIN                   | 26 |
| 27 |                                                            | STORE | THETA                  | 27 |
| 28 |                                                            | PDDL  | BMN                    | 28 |
| 29 |                                                            |       | COSTH                  | 29 |
| 30 |                                                            | DLOAD | NEGCOS                 | 30 |
| 31 | NEGCOS                                                     | DLOAD | RVQ                    | 31 |
| 32 |                                                            | BPL   | DCOMP                  | 32 |
| 33 |                                                            |       | DAD                    | 33 |
| 34 |                                                            |       | NEGOUT                 | 34 |
| 35 | ARCTANXX                                                   | STORE | DP1/2                  | 35 |
| 36 |                                                            | RVQ   | THETA                  | 36 |
| 37 |                                                            |       |                        | 37 |
| 38 | NEGOUT                                                     | DSU   | GOTO                   | 38 |
| 39 |                                                            |       | DP1/2                  | 39 |
| 40 |                                                            |       | ARCTANXX               | 40 |
| 41 | ATAN=90                                                    | DLOAD | SIGN                   | 41 |
| 42 |                                                            |       | LODP1/4                | 42 |
| 43 |                                                            |       | SINTH                  | 43 |
| 44 |                                                            | STORE | THETA                  | 44 |
| 45 |                                                            | RVQ   |                        | 45 |
| 46 |                                                            |       |                        | 46 |
| 47 | 2DZERO                                                     | =     | DPZERO                 | 47 |
| 48 |                                                            |       |                        | 48 |
| 49 |                                                            |       |                        | 49 |
| 50 |                                                            |       |                        | 50 |
| 51 |                                                            |       |                        | 51 |
| 52 |                                                            |       |                        | 52 |
| 53 |                                                            |       |                        | 53 |
| 54 |                                                            |       |                        | 54 |
| 55 |                                                            |       |                        | 55 |
| 56 |                                                            |       |                        | 56 |
| 57 |                                                            |       |                        | 57 |
| 58 |                                                            |       |                        | 58 |
| 59 |                                                            |       |                        | 59 |
| 60 |                                                            |       |                        | 60 |

# ..... SETGAMMA SUBROUTINE .....  
# SUBROUTINE TO SET GAMMA FOR THE LAT-LONG AND LALOTORV SUBROUTINES

#  
# GAMMA = B\*\*2/A\*\*2 FOR EARTH (B-1)  
# GAMMA = 1 FOR MOON (B-1)

#  
# CALLING SEQUENCE  
# L        CALL

# L+1                SETGAMMA  
#  
# INPUT

# LUNAFLAG=0 FOR EARTH,=1 FOR MOON  
#  
# OUTPUT

# GAMMA IN GAMRP (B-1)

SETGAMMA            DLOAD    BOFF            # BRANCH FOR EARTH  
                              B2/A2            # EARTH GAMMA  
                              LUNAFLAG  
                              SETGMEX

                      SLOAD            1B1            # MOON GAMMA  
SETGMEX            STORE    GAMRP

GAMRP            RVQ        8D

```
.....SETRE SUBROUTINE
SUBROUTINE TO SET RE (EARTH OR MOON RADIUS)
#
RE= RM FOR MOON
RE= RREF FOR FIXED EARTH RADIUS OR COMPUTED RF FOR FISCHER ELLIPSOID
#
CALLING SEQUENCE
L CALL
#
L+1 SETRE
#
SUBROUTINES USED
GETERAD
#
INPUT
ERADFLAG=0 FOR FIXED RE, 1 FOR COMPUTED RE
ALPHAV +4= 1/2 SINL IF GETERAD IS CALLED
LUNAFLAG=0 FOR EARTH,=1 FOR MOON
#
OUTPUT
ERADM= 504RM FOR MOON (METERS B-29)
ERADM= ERAD OR COMPUTED RE FOR EARTH (METERS B-29)
```

```
SETRE STQ DLOAD
 SETREX
 504RM
 BON DLOAD # BRANCH FOR MOON
 LUNAFLAG
 TSTRLSRM
 ERAD
 BOFF CALL # ERADFLAG=0 FOR FIXED RE,1 FOR COMPUTED
 ERADFLAG
 SETRXX
 GETERAD
SETRXX STCALL ERADM # EXIT WITH RE OR RM METERS B-29
 SETREX
TSTRLSRM BON VLOAD # ERADFLAG=0,SET R0=RLS
 ERADFLAG # =1 R0=RM
 SETRXX
 RLS
 ABVAL SR2R # SCALE FROM B-27 TO B-29
 GOTO
SETREX = SETRXX
 S2
```

```
1 # RP-TO-R SUBROUTINE
2 # SUBROUTINE TO CONVERT RP (VECTOR IN PLANETARY COORDINATE SYSTEM,EITHER
3 # EARTH-FIXED OR MOON-FIXED) TO R (SAME VECTOR IN THE BASIC REF. SYSTEM)
4 #
5 # R=MT(T)*(RP+LPXRP) MT= M MATRIX TRANSPOSE
6 #
7 # CALLING SEQUENCE
8 # L CALL
9 # L+1 RP-TO-R
10 #
11 # SUBROUTINES USED
12 # EARTHMX,MOONMX,EARTHLM
13 #
14 # ITEMS AVAILABLE FROM LAUNCH DATA
15 # 504LM= THE LIBRATION VECTOR L OF THE MOON AT TIME TIMSUBL,EXPRESSED
16 # IN THE MOON-FIXED COORD. SYSTEM RADIANS B0
17 # ITEMS NECESSARY FOR SUBR. USED (SEE DESCRIPTION OF SUBR.)
18 #
19 # INPUT
20 # MPAC= 0 FOR EARTH, NON-ZERO FOR MOON
21 # 0-5D= RP VECTOR
22 # 6-7D= TIME
23 #
24 # OUTPUT
25 # MPAC= R VECTOR METERS B-29 FOR EARTH, B-27 FOR MOON
26
27
28 SETLOC PLANTINI
29 BANK
30
31 COUNT* $$/LUR0T
32
33 RP-TO-R STQ BHIZ
34 RPREXIT
35 RPTORA
36
37 CALL # COMPUTE M MATRIX FOR MOON
38 MOONMX # LP=LM FOR MOON RADIANS B0
39
40 VLOAD
41 504LM
42 RPTORB VXV VAD
43 504RPR
44 504RPR
45
46 VXM GOTO
47 MMATRIX # MPAC=R=MT(T)*(RP+LPXRP)
48 RPRPXXXX # RESET PUSHLOC TO 0 BEFORE EXITING
49
50 RPTORA CALL
51 EARTHMX # EARTH COMPUTATIONS
52 # M MATRIX B-1
53
54 CALL
55 EARTHLM # L VECTOR RADIANS B0
56 VSL1 # LP=M(T)*L RAD B-0
57 MMATRIX
```



|    |        |            |    |
|----|--------|------------|----|
| 1  |        |            | 1  |
| 2  | GOTO   |            | 2  |
| 3  |        | RPTORB     | 3  |
| 4  | SETLOC | PLANTIN    | 4  |
| 5  | BANK   |            | 5  |
| 6  | COUNT* | \$\$/LUR0T | 6  |
| 7  |        |            | 7  |
| 8  |        |            | 8  |
| 9  |        |            | 9  |
| 10 |        |            | 10 |
| 11 |        |            | 11 |
| 12 |        |            | 12 |
| 13 |        |            | 13 |
| 14 |        |            | 14 |
| 15 |        |            | 15 |
| 16 |        |            | 16 |
| 17 |        |            | 17 |
| 18 |        |            | 18 |
| 19 |        |            | 19 |
| 20 |        |            | 20 |
| 21 |        |            | 21 |
| 22 |        |            | 22 |
| 23 |        |            | 23 |
| 24 |        |            | 24 |
| 25 |        |            | 25 |
| 26 |        |            | 26 |
| 27 |        |            | 27 |
| 28 |        |            | 28 |
| 29 |        |            | 29 |
| 30 |        |            | 30 |
| 31 |        |            | 31 |
| 32 |        |            | 32 |
| 33 |        |            | 33 |
| 34 |        |            | 34 |
| 35 |        |            | 35 |
| 36 |        |            | 36 |
| 37 |        |            | 37 |
| 38 |        |            | 38 |
| 39 |        |            | 39 |
| 40 |        |            | 40 |
| 41 |        |            | 41 |
| 42 |        |            | 42 |
| 43 |        |            | 43 |
| 44 |        |            | 44 |
| 45 |        |            | 45 |
| 46 |        |            | 46 |
| 47 |        |            | 47 |
| 48 |        |            | 48 |
| 49 |        |            | 49 |
| 50 |        |            | 50 |
| 51 |        |            | 51 |
| 52 |        |            | 52 |
| 53 |        |            | 53 |
| 54 |        |            | 54 |
| 55 |        |            | 55 |
| 56 |        |            | 56 |
| 57 |        |            | 57 |
| 58 |        |            | 58 |
| 59 |        |            | 59 |
| 60 |        |            | 60 |

```
1 # R-TO-RP SUBROUTINE
2 # SUBROUTINE TO CONVERT R (VECTOR IN REFERENCE COORD. SYSTEM) TO RP
3 # (VECTOR IN PLANETARY COORD SYSTEM) EITHER EARTH-FIXED OR MOON-FIXED
4 #
5 # RP=M(T)*(R-LXR)
6 #
7 # CALLING SEQUENCE
8 # L CALL
9 # L+1 R-TO-RP
10 #
11 # SUBROUTINES USED
12 # EARTHMX,MOONMX,EARTHHL
13 #
14 # INPUT
15 # MPAC= 0 FOR EARTH, NON-ZERO FOR MOON
16 # 0-5D= R VECTOR
17 # 6-7D= TIME
18 #
19 # ITEMS AVAILABLE FROM LAUNCH DATA
20 # 504LM= THE LIBRATION VECTOR L OF THE MOON AT TIME TIMSUBL,EXPRESSED
21 # IN THE MOON-FIXED COORD. SYSTEM RADIANS B0
22 # ITEMS NECESSARY FOR SUBROUTINES USED (SEE DESCRIPTION OF SUBR.)
23 #
24 # OUTPUT
25 # MPAC=RP VECTOR METERS B-29 FOR EARTH, B-27 FOR MOON
26
27 R-TO-RP STQ BHIZ
28 RPREXIT
29 RTORPA
30
31 CALL
32 MOONMX
33 VLOAD VXM
34 504LM # LP=LM
35 MMATRIX
36 VSL1 # L=MT(T)*LP RADIANS B0
37 RTORPB VXV BVSU
38 504RPR
39 504RPR
40 MXV # M(T)*(R-LXR) B-2
41 RPRPXXXX VSL1 MMATRIX
42 SETPD
43 OD
44 GOTO
45 RPREXIT
46 RTORPA CALL # EARTH COMPUTATIONS
47 EARTHMX
48 CALL
49 EARTHHL
50 GOTO # MPAC=L=(-AX,-AY,0) RAD B-0
51 RTORPB
```

```
1 # MOONMX SUBROUTINE
2 # SUBROUTINE TO COMPUTE THE TRANSFORMATION MATRIX M FOR THE MOON
3
4 #
5 # CALLING SEQUENCE
6 # L CALL
7 # L+1 MOONMX
8 #
9 # SUBROUTINES USED
10 # NEWANGLE
11 #
12 # INPUT
13 # 6-7D= TIME
14 # ITEMS AVAILABLE FROM LAUNCH DATA
15 # BSUBO,BDOT
16 # TIMSUBO,NODIO,NODDOT,FSUBO,FDOT
17 # COSI= COS(I) B-1
18 # SINI= SIN(I) B-1
19 # I IS THE ANGLE BETWEEN THE MEAN LUNAR EQUATORIAL PLANE AND THE
20 # PLANE OF THE ECLIPTIC (1 DEGREE 32.1 MINUTES)
21 #
22 # OUTPUT
23 # MMATRIX= 3X3 M MATRIX B-1 (STORED IN VAC AREA)
24
25 MOONMX STQ SETPD
26 EARTHMX
27 8D
28
29 AXT,1 # B REQUIRES SL 0, SL 5 IN NEWANGLE
30
31 DLOAD 5
32 PDDL # PD 10D 8-9D=BSUBO
33 BSUBO # 10-11D= BDOT
34 BDOT
35 CALL # PD 12D
36 NEWANGLE # EXIT WITH PD 8D AND MPAC= B REVS B0
37 PUSH COS
38 STODL COB
39 SIN # PD 10D
40 STODL COB
41 SOB # PD 8D COS(B) B-1
42 FSUBO # PD 8D SIN(B) B-1
43 FSUBO # SETUP INPUT FOR NEWANGLE
44 FSUBO # 8-9D=FSUBO
45 PDDL # PD 10D THEN 12D 10-11D=FDOT
46 PUSH FDOT
47 AXT,1 CALL
48 CALL # F REQUIRES SL 1, SL 6 IN NEWANGLE
49 4
50 NEWANGLE # EXIT WITH PD 8D AND MPAC= F REVS B0
51 STODL AVECTR +2
52 AVECTR # SAVE F TEMP
53 NODIO #
54 PDDL # PD 10D THEN 12D 8-9D=NODIO
55 PUSH # PD 10D THEN 12D 10-11D=NODDOT
56 NODDOT # MPAC=T
57 NODDOT #
58 AXT,1 CALL
59 5
60 NEWANGLE # NODE REQUIRES SL 0, SL 5 IN NEWANGLE
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```



|    |   |
|----|---|
| 76 | 1 |
| 77 |   |

```
1
2 DVECTR
3 PDDL SIN # PD 20D 14-19D= DVECTR*COSF B-2
4 504F
5 VXSC VSU # PD 14D AVECTR*SINF B-2
6 AVECTR
7 VSL1
8 STODL MMATRIX +6 # M1= AVECTR*SINF-DVECTR*COSF B-1
9 504F
10 SIN VXSC # PD 8D
11 PDDL COS # PD 14D 8-13D=DVECTR*SINF B-2
12 504F
13 VXSC VAD # PD 8D AVECTR*COSF B-2
14 AVECTR
15 VSL1 VCOMP
16 STCALL MMATRIX # M0= -(AVECTR*COSF+DVECTR*SINF) B-1
17 EARTHMX
18 # COMPUTE X=X0+(XDOT)(T+T0)
19 # 8-9D= X0 (REVS B-0),PUSHLOC SET AT 12D
20 # 10-11D=XDOT (REVS/CSEC) SCALED B+23 FOR WEARTH,B+28 FOR NODDOT AND BDOT
21 # AND B+27 FOR FDOT
22 # X1=DIFFERENCE IN 23 AND SCALING OF XDOT,=0 FOR WEARTH,5 FOR NODDOT AND
23 # BDOT AND 4 FOR FDOT
24 # 6-7D=T (CSEC B-28), TIMSUB0= (CSEC B-42 TRIPLE PREC.)
25
26 NEWANGLE DLOAD SR # ENTER PD 12D
27 6D
28 14D
29 TAD TLOAD # CHANGE MODE TO TP
30 TIMSUB0
31 MPAC
32 STODL TIMSUBM # T+T0 CSEC B-42
33 TIMSUBM +1
34 DMP
35 SL* DAD # PD 10D MULT BY XDOT IN 10-11D
36 5,1 # PD 8D ADD X0 IN 8-9D AFTER SHIFTING
37 PUSH SLOAD # SUCH THAT SCALING IS B-0
38 TIMSUBM # PD 10D SAVE PARTIAL (X0+XDOT*T) IN 8-9D
39 SL DMP
40 9D
41 10D # XDOT
42 SL* DAD # PD 8D SHIFT SUCH THAT THIS PART OF X
43 10D,1 # IS SCALED REVS/CSEC B-0
44 BOV # TURN OFF OVERFLOW IF SET BY SHIFT
45 +1 # INSTRUCTION BEFORE EXITING
46 RVQ # MPAC=X= X0+(XDOT)(T+T0) REVS B0
```

```
..... EARTHMX SUBROUTINE
SUBROUTINE TO COMPUTE THE TRANSFORMATION MATRIX M FOR THE EARTH
#
CALLING SEQUENCE
L CALL
L+1 EARTHMX
#
SUBROUTINE USED
NEWANGLE
#
INPUT
INPUT AVAILABLE FROM LAUNCH DATA AZO REVS B-0
TEPHEM CSEC B-42
6-7D= TIME CSEC B-28
#
OUTPUT
MMATRIX= 3X3 M MATRIX B-1 (STORED IN VAC AREA)
#
BANK 26
SETLOC PLANTIN1
BANK
COUNT* $$/LUR0T
#
EARTHMX STQ SETPD # SET 8-9D=AZO
 EARTHMX 8D # 10-11D=WEARTH
 AXT,1 0 # FOR SL 5, AND SL 10 IN NEWANGLE
 DLOAD PDDL # LEAVING PD SET AT 12D FOR NEWANGLE
 AZO
 WEARTH
 PUSH CALL
 NEWANGLE
 SETPD PUSH # 18-19D=504AZ
 18D #
 COS PDDL # 20-37D= MMATRIX= COS(AZ) SIN(AZ) 0
 504AZ # -SIN(AZ) COS(AZ) 0 B-1
 SIN PDDL # 0 0 1
 HI6ZEROS
 PDDL SIN
 504AZ
 DCOMP PDDL
 504AZ
 COS PDVL
 HI6ZEROS
 PDDL PUSH
 HIDPHALF
 GOTO
 EARTHMX
```

1412THE



# PLANETARY\_INERTIAL\_ORIENTATION

```
..... EARTHL SUBROUTINE
SUBROUTINE TO COMPUTE L VECTOR FOR EARTH
#
CALLING SEQUENCE
L CALL
L+1 EARTHL
#
INPUT
AXO,AYO SET AT LAUNCH TIME WITH AYO IMMEDIATELY FOLLOWING AXO IN CORE
#
OUTPUT
MPAC= -AX RADIANS B-0
-AY
0
#
BANK 06
SETLOC EARTHLOC
BANK
COUNT* $$/LUR0T
#
EARTHL DLOAD DCOMP
AXO
STODL 504LPL
-AYO
STODL 504LPL +2
L06ZEROS
STOVL 504LPL +4
504LPL
RVQ
```

|    |                                      |   |       |                                     |    |
|----|--------------------------------------|---|-------|-------------------------------------|----|
| 1  | # CONSTANTS AND ERASABLE ASSIGNMENTS |   |       |                                     | 1  |
| 2  | 1B1                                  | = | DP1/2 | # 1 SCALED B-1                      | 2  |
| 3  | RPREXIT                              | = | S1    | # R-TO-RP AND RP-TO-R SUBR EXIT     | 3  |
| 4  | EARTHMX                              | = | S2    | # EARTHMX,MOONMX SUBR. EXITS        | 4  |
| 5  | 504RPR                               | = | 0D    | # 6 REGS R OR RP VECTOR             | 5  |
| 6  | SINNODI                              | = | 8D    | # 2 SIN(NODI)                       | 6  |
| 7  | DVECTR                               | = | 8D    | # 6 D VECTOR MOON                   | 7  |
| 8  | CVECTR                               | = | 8D    | # 6 C VECTR MOON                    | 8  |
| 9  | 504AZ                                | = | 18D   | # 2 AZ                              | 9  |
| 10 | TIMSUBM                              | = | 14D   | # 3 TIME SUB M (MOON) T+T0 IN GETAZ | 10 |
| 11 | 504LPL                               | = | 14D   | # 6 L OR LP VECTOR                  | 11 |
| 12 | AVECTR                               | = | 20D   | # 6 A VECTOR (MOON)                 | 12 |
| 13 | BVECTR                               | = | 26D   | # 6 B VECTOR (MOON)                 | 13 |
| 14 | MMATRIX                              | = | 20D   | # 18 M MATRIX                       | 14 |
| 15 | COB                                  | = | 32D   | # 2 COS(B) B-1                      | 15 |
| 16 | SOB                                  | = | 34D   | # 2 SIN(B) B-1                      | 16 |
| 17 | 504F                                 | = | 6D    | # 2 F (MOON)                        | 17 |
| 18 |                                      |   |       |                                     | 18 |
| 19 |                                      |   |       |                                     | 19 |
| 20 |                                      |   |       |                                     | 20 |
| 21 |                                      |   |       |                                     | 21 |
| 22 |                                      |   |       |                                     | 22 |
| 23 |                                      |   |       |                                     | 23 |
| 24 |                                      |   |       |                                     | 24 |
| 25 |                                      |   |       |                                     | 25 |
| 26 |                                      |   |       |                                     | 26 |
| 27 |                                      |   |       |                                     | 27 |
| 28 |                                      |   |       |                                     | 28 |
| 29 |                                      |   |       |                                     | 29 |
| 30 |                                      |   |       |                                     | 30 |
| 31 |                                      |   |       |                                     | 31 |
| 32 |                                      |   |       |                                     | 32 |
| 33 |                                      |   |       |                                     | 33 |
| 34 |                                      |   |       |                                     | 34 |
| 35 |                                      |   |       |                                     | 35 |
| 36 |                                      |   |       |                                     | 36 |
| 37 |                                      |   |       |                                     | 37 |
| 38 |                                      |   |       |                                     | 38 |
| 39 |                                      |   |       |                                     | 39 |
| 40 |                                      |   |       |                                     | 40 |
| 41 |                                      |   |       |                                     | 41 |
| 42 |                                      |   |       |                                     | 42 |
| 43 |                                      |   |       |                                     | 43 |
| 44 |                                      |   |       |                                     | 44 |
| 45 |                                      |   |       |                                     | 45 |
| 46 |                                      |   |       |                                     | 46 |
| 47 |                                      |   |       |                                     | 47 |
| 48 |                                      |   |       |                                     | 48 |
| 49 |                                      |   |       |                                     | 49 |
| 50 |                                      |   |       |                                     | 50 |
| 51 |                                      |   |       |                                     | 51 |
| 52 |                                      |   |       |                                     | 52 |
| 53 |                                      |   |       |                                     | 53 |
| 54 |                                      |   |       |                                     | 54 |
| 55 |                                      |   |       |                                     | 55 |
| 56 |                                      |   |       |                                     | 56 |
| 57 |                                      |   |       |                                     | 57 |
| 58 |                                      |   |       |                                     | 58 |
| 59 |                                      |   |       |                                     | 59 |
| 60 |                                      |   |       |                                     | 60 |

```
INCORP1 -- PERFORMS THE SIX DIMENSIONAL STATE VECTOR DEVIATION FOR POSITION
AND VELOCITY OR THE NINE-DIMENSIONAL DEVIATION OF POSITION, VELOCITY, AND
RADAR OR LANDMARK BIAS. THE OUTPUT OF THE BVECTOR ROUTINE ALONG WITH THE
ERROR TRANSITION MATRIX (W) ARE USED AS INPU TO THE ROUTINE. THE DEVIATION
IS OBTAINED BY COMPUTING AN ESTIMATED TRACKING MEASUREMENT FROM THE
CURRENT STATE VECTOR AND COMPARING IT WITH AN ACTUAL TRACKING MEASUREMENT
AND APPLYING A STATISTICAL WEIGHTING VECTOR.
```

## # INPUT

```
DMENFLG = 0 (6-DIMENSIONAL BVECTOR), =1 (9-DIMENSIONAL)
W = ERROR TRANSITION MATRIX 6X6 OR 9X9
VARIANCE = VARIANCE (SCALAR)
DELTAQ = MEASURED DEVIATION (SCALAR)
BVECTOR = 6 OR 9 DIMENSIONAL BVECTOR
```

## # OUTPUT

```
DELTAX = STATE VECTOR DEVIATIONS 6 OR 9 DIMENSIONAL
ZI = VECTOR USED FOR THE INCORPORATION 6 OR 9 DIMENSIONAL
GAMMA = SCALAR
OMEGA = OMEGA WEIGHTING VECTOR 6 OR 9 DIMENSIONAL
```

## # CALLING SEQUENCE

```
L CALL INCORP1
```

## # NORMAL EXIT

```
L+1 OF CALLING SEQUENCE
```

```
 BANK 37
 SETLOC MEASINC
 BANK
```

```
 COUNT* $$/INCOR
```

```
 EBANK= W
```

INCORP1

STQ

```
 AXT,1 EGRESS
 SSP
```

54D

S1

18D

# IX1 = 54

S1= 18

AXT,2

SSP

18D

S2

6

# IX2 = 18

S2=6

Z123

VLOAD

MXV\*

BVECTOR

# BVECTOR (0)

W +54D,1

STORE

ZI +18D,2

VLOAD

BVECTOR +6

# BVECTOR (1)

[illegible]

|    |                               |       |                     |    |
|----|-------------------------------|-------|---------------------|----|
| 1  | # MEASUREMENT - INCORPORATION |       |                     | 1  |
| 2  |                               | DMP   | SQRT                | 2  |
| 3  |                               |       | TEMPVAR             | 3  |
| 4  |                               | SL*   | TAD                 | 4  |
| 5  |                               |       | 0,2                 | 5  |
| 6  |                               |       | TRIPA               | 6  |
| 7  |                               | NORM  | INCR,2              | 7  |
| 8  |                               |       | X2                  | 8  |
| 9  |                               | DEC   | -2                  | 9  |
| 10 |                               | SXA,2 | AXT,2               | 10 |
| 11 |                               |       | NORMGAM             | 11 |
| 12 |                               |       | 162D                | 12 |
| 13 |                               | BDDV  | SETPD               | 13 |
| 14 |                               |       | DP1/4TH             | 14 |
| 15 |                               |       | 0                   | 15 |
| 16 |                               | STORE | GAMMA               | 16 |
| 17 |                               | TLOAD | NORM                | 17 |
| 18 |                               |       | TRIPA               | 18 |
| 19 |                               |       | X1                  | 19 |
| 20 |                               | DLOAD | PDDL                | 20 |
| 21 |                               |       | MPAC                | 21 |
| 22 |                               |       | DELTAQ              | 22 |
| 23 |                               | NORM  |                     | 23 |
| 24 |                               |       | S1                  | 24 |
| 25 |                               | XSU,1 | SR1                 | 25 |
| 26 |                               |       | S1                  | 26 |
| 27 |                               | DDV   | PUSH                | 27 |
| 28 |                               | GOTO  |                     | 28 |
| 29 |                               |       | NEWZCOMP            | 29 |
| 30 | -3                            | SSP   |                     | 30 |
| 31 |                               |       | S2                  | 31 |
| 32 |                               |       | 54D                 | 32 |
| 33 | INCOR2                        | VLOAD | VXM*                | 33 |
| 34 |                               |       | # COMPUT OMEGA1,2,3 | 34 |
| 35 |                               |       | ZI                  | 35 |
| 36 |                               | PUSH  | W +162D,2           | 36 |
| 37 |                               |       | VLOAD               | 37 |
| 38 |                               |       | ZI +6               | 38 |
| 39 |                               | VXM*  | VAD                 | 39 |
| 40 |                               |       | W +180D,2           | 40 |
| 41 |                               | PUSH  | VLOAD               | 41 |
| 42 |                               |       | ZI +12D             | 42 |
| 43 |                               | VXM*  | VAD                 | 43 |
| 44 |                               |       | W +198D,2           | 44 |
| 45 |                               | PUSH  | TIX,2               | 45 |
| 46 |                               |       | INCOR2              | 46 |
| 47 |                               | VLOAD | STADR               | 47 |
| 48 |                               | STORE | OMEGA +12D          | 48 |
| 49 |                               | VLOAD | STADR               | 49 |
| 50 |                               | STORE | OMEGA +6            | 50 |
| 51 |                               | VLOAD | STADR               | 51 |
| 52 |                               | STORE | OMEGA               | 52 |
| 53 |                               |       |                     | 53 |
| 54 |                               |       |                     | 54 |
| 55 |                               |       |                     | 55 |
| 56 |                               |       |                     | 56 |
| 57 |                               |       |                     | 57 |
| 58 |                               |       |                     | 58 |
| 59 |                               |       |                     | 59 |
| 60 |                               |       |                     | 60 |



| # MEASUREMENT INCORPORATION |          |              | PAGE 112 |
|-----------------------------|----------|--------------|----------|
| 1                           |          |              | 1        |
| 2                           |          | BON          | 2        |
| 3                           |          | VLOAD        | 3        |
| 4                           |          | DMENFLG      | 4        |
| 5                           |          | INCOR2AB     | 5        |
| 6                           |          | ZEROVECS     | 6        |
| 7                           |          | OMEGA +12D   | 7        |
| 8                           | INCOR2AB | AXT,2        | 8        |
| 9                           |          | SSP          | 9        |
| 10                          |          | 18D          | 10       |
| 11                          |          | S2           | 11       |
| 12                          |          | 6            | 12       |
| 13                          | INCOR3   | VLOAD*       | 13       |
| 14                          |          | OMEGA +18D,2 | 14       |
| 15                          |          | VXSC         | 15       |
| 16                          |          | VSL*         | 16       |
| 17                          |          | 0            | 17       |
| 18                          |          | # DELTAQ/A   | 18       |
| 19                          |          | 0,1          | 19       |
| 20                          |          | DELTA +18D,2 | 20       |
| 21                          |          | VLOAD        | 21       |
| 22                          |          | INCOR3       | 22       |
| 23                          |          | DELTA +6     | 23       |
| 24                          |          | VSL3         | 24       |
| 25                          |          | STORE        | 25       |
| 26                          |          | DELTA +6     | 26       |
| 27                          |          | GOTO         | 27       |
| 28                          |          | EGRESS       | 28       |
| 29                          |          |              | 29       |
| 30                          |          |              | 30       |
| 31                          |          |              | 31       |
| 32                          |          |              | 32       |
| 33                          |          |              | 33       |
| 34                          |          |              | 34       |
| 35                          |          |              | 35       |
| 36                          |          |              | 36       |
| 37                          |          |              | 37       |
| 38                          |          |              | 38       |
| 39                          |          |              | 39       |
| 40                          |          |              | 40       |
| 41                          |          |              | 41       |
| 42                          |          |              | 42       |
| 43                          |          |              | 43       |
| 44                          |          |              | 44       |
| 45                          |          |              | 45       |
| 46                          |          |              | 46       |
| 47                          |          |              | 47       |
| 48                          |          |              | 48       |
| 49                          |          |              | 49       |
| 50                          |          |              | 50       |
| 51                          |          |              | 51       |
| 52                          |          |              | 52       |
| 53                          |          |              | 53       |
| 54                          |          |              | 54       |
| 55                          |          |              | 55       |
| 56                          |          |              | 56       |
| 57                          |          |              | 57       |
| 58                          |          |              | 58       |
| 59                          |          |              | 59       |
| 60                          |          |              | 60       |

# INCORP2 - INCORPORATES THE COMPUTED STATE VECTOR DEVIATIONS INTO THE  
# ESTIMATED STATE VECTOR. THE STATE VECTOR UPDATED MAY BE FOR EITHER THE  
# LEM OR THE CSM. DETERMINED BY FLAG VEHUPFLG. (ZERO = LEM) (1 = CSM)

# INPUT

# PERMANENT STATE VECTOR FOR EITHER THE LEM OR CSM  
# VEHUPFLG = UPDATE VEHICLE 0=LEM 1=CSM  
# W = ERROR TRANSITION MATRIX  
# DELTAX = COMPUTED STATE VECTOR DEVIATIONS  
# DMENFLG = SIZE OF W MATRIX (ZERO=6X6) (1=9X9)  
# GAMMA = SCALAR FOR INCORPORATION  
# ZI = VECTOR USED IN INCORPORATION  
# OMEGA = WEIGHTING VECTOR

# OUTPUT

# UPDATED PERMANENT STATE VECTOR

# CALLING SEQUENCE

# L CALL INCORP2

# NORMAL EXIT

# L+1 OF CALLING SEQUENCE

SETLOC MEASINC1  
BANK

COUNT\* \$\$/INCOR

INCORP2 STQ CALL  
EGRESS  
INTSTALL

VLOAD VXSC # CALC. GAMMA \* OMEGA1,2,3  
OMEGA  
GAMMA

STOVL OMEGAM1  
OMEGA +6

VXSC  
STOVL GAMMA  
OMEGAM2  
OMEGA +12D

VXSC  
STORE GAMMA  
OMEGAM3

EXIT  
CAF 54DD # INITIAL IX 1 SETTING FOR W MATRIX  
TS WIXA

TS WIXB  
CAF ZERO  
TS ZIXA # INITIAL IX 2 SETTING FOR Z COMPONENT

FAZA TS ZIXB  
TC PHASCHNG

FAZA1

OCT  
TC  
ADRES  
CA  
TS  
CA  
TS  
TC

04022  
UPFLAG  
REINTFLG  
WIXB  
WIXA  
ZIXB  
ZIXA  
INTPRET

# START FIRST PHASE OF INCORP2  
# TO UPDATE 6 OR 9 DIM. W MATRIX IN TEMP

LXA,1

LXA,2  
WIXA  
ZIXA

SSP

DLOAD\*  
S1  
6

DCOMP

ZI,2  
NORM  
S2

# CALC UPPER 3X9 PARTITION OF W MATRIX

VXSC

XCHX,2  
OMEGAM1  
S2

LXC,2

XAD,2  
X2  
NORMGAM

VSL\*

XCHX,2  
0,2  
S2

VAD\*

W +54D,1  
STORE  
HOLDW

DLOAD\*

DCOMP  
ZI,2  
VXSC

# CALC MIDDLE 3X9 PARTITION OF W MATRIX

NORM

S2  
OMEGAM2  
LXC,2

XCHX,2

S2  
X2  
VSL\*

XAD,2

NORMGAM  
0,2  
VAD\*

XCHX,2

S2  
W +108D,1  
STORE  
HOLDW +6

BOFF

DMENFLG  
FAZB

# BRANCH IF 6 DIMENSIONAL

DLOAD\*

DCOMP  
ZI,2  
VXSC

# CALC LOWER 3X9 PARTITION OF W MATRIX

NORM

1

|    |                              |        |                                   |    |
|----|------------------------------|--------|-----------------------------------|----|
| 1  | # MEASUREMENT_INCREMENTATION |        |                                   | 1  |
| 2  |                              | VLOAD  | VAD                               | 2  |
| 3  |                              |        | X789                              | 3  |
| 4  |                              |        | DELTAX +12D                       | 4  |
| 5  |                              | STORE  | TX789                             | 5  |
| 6  |                              | BON    | RTB                               | 6  |
| 7  |                              |        | VEHUPFLG                          | 7  |
| 8  |                              |        | DOCSM                             | 8  |
| 9  |                              |        | MOVEPLEM                          | 9  |
| 10 | FAZAB                        | BOVB   | AXT,2                             | 10 |
| 11 |                              |        | TCDANZIG                          | 11 |
| 12 |                              |        | 0                                 | 12 |
| 13 |                              | BOFF   | AXT,2                             | 13 |
| 14 |                              |        | MOONTHIS                          | 14 |
| 15 |                              |        | +2                                | 15 |
| 16 |                              |        | 2                                 | 16 |
| 17 |                              | VLOAD  | VSR*                              | 17 |
| 18 |                              |        | DELTAX                            | 18 |
| 19 |                              |        | # B27 IF MOON ORBIT, B29 IF EARTH | 19 |
| 20 |                              | VAD    | 0 -7,2                            | 20 |
| 21 |                              |        | BOV                               | 21 |
| 22 |                              |        | TDELTAV                           | 22 |
| 23 |                              | STOVL  | FAZAB1                            | 23 |
| 24 |                              |        | TDELTAV                           | 24 |
| 25 |                              |        | DELTAX +6                         | 25 |
| 26 |                              |        | # B5 IF MOON ORBIT, B7 IF EARTH   | 26 |
| 27 |                              | VSR*   | VAD                               | 27 |
| 28 |                              |        | 0 -4,2                            | 28 |
| 29 |                              |        | TNUV                              | 29 |
| 30 |                              | BOV    |                                   | 30 |
| 31 |                              | STCALL | FAZAB2                            | 31 |
| 32 |                              |        | TNUV                              | 32 |
| 33 | FAZAB1                       | VLOAD  | FAZAB3                            | 33 |
| 34 |                              |        | VAD                               | 34 |
| 35 |                              |        | RCV                               | 35 |
| 36 | FAZAB2                       | STORE  | DELTAX                            | 36 |
| 37 |                              | VLOAD  | RCV                               | 37 |
| 38 |                              |        | VAD                               | 38 |
| 39 |                              |        | VCV                               | 39 |
| 40 |                              |        | DELTAX +6                         | 40 |
| 41 |                              | STORE  | VCV                               | 41 |
| 42 |                              | SXA,2  | CALL                              | 42 |
| 43 |                              |        | PBODY                             | 43 |
| 44 | FAZAB3                       | CALL   | RECTIFY                           | 44 |
| 45 |                              |        |                                   | 45 |
| 46 |                              | BON    | GRP2PC                            | 46 |
| 47 |                              |        | RTB                               | 47 |
| 48 |                              |        | VEHUPFLG                          | 48 |
| 49 |                              |        | DOCSM1                            | 49 |
| 50 |                              |        | MOVEALEM                          | 50 |
| 51 | FAZAB4                       | CALL   | SVDWN2                            | 51 |
| 52 |                              |        | # STORE DOWNLINK STATE VECTOR     | 52 |
| 53 |                              |        |                                   | 53 |
| 54 |                              |        |                                   | 54 |
| 55 |                              |        |                                   | 55 |
| 56 |                              |        |                                   | 56 |
| 57 |                              |        |                                   | 57 |
| 58 |                              |        |                                   | 58 |
| 59 |                              |        |                                   | 59 |
| 60 |                              |        |                                   | 60 |

|    |          |        |            |                               |    |
|----|----------|--------|------------|-------------------------------|----|
| 1  |          |        | GRP2PC     | # PHASE CHANGE                | 1  |
| 2  |          | BOFF   | VLOAD      |                               | 2  |
| 3  |          |        | DMENFLG    |                               | 3  |
| 4  |          |        | FAZAB5     | # 6 DIMENSIONAL               | 4  |
| 5  |          |        | TX789      | # 9 DIMENSIONAL               | 5  |
| 6  |          |        | X789       |                               | 6  |
| 7  | FAZAB5   | STORE  | SXA,1      |                               | 7  |
| 8  |          | LXA,1  | EGRESS     |                               | 8  |
| 9  |          |        | QPRET      |                               | 9  |
| 10 |          | EXIT   |            |                               | 10 |
| 11 |          | TC     | POSTJUMP   | # EXIT                        | 11 |
| 12 |          | CADR   | INTWAKE    |                               | 12 |
| 13 | DOCSM    | RTB    | GOTO       |                               | 13 |
| 14 |          |        | MOVEPCSM   |                               | 14 |
| 15 |          |        | FAZAB      |                               | 15 |
| 16 | DOCSM1   | RTB    | CALL       |                               | 16 |
| 17 |          |        | MOVEACSM   |                               | 17 |
| 18 |          |        | SVDWN1     | # STORE DOWNLINK STATE VECTOR | 18 |
| 19 |          | GOTO   |            |                               | 19 |
| 20 |          |        | FAZAB4     |                               | 20 |
| 21 | ZEROD    | =      | ZEROVECS   |                               | 21 |
| 22 | 54DD     | DEC    | 54         |                               | 22 |
| 23 | 6DD      | DEC    | -6         |                               | 23 |
| 24 | 12DD     | DEC    | 12         |                               | 24 |
| 25 |          | SETLOC | RENDEZ     |                               | 25 |
| 26 |          | BANK   |            |                               | 26 |
| 27 |          | COUNT* | \$\$/INCOR |                               | 27 |
| 28 |          |        |            |                               | 28 |
| 29 | NEWZCOMP | VLOAD  | ABVAL      |                               | 29 |
| 30 |          |        | ZI         |                               | 30 |
| 31 |          | STOVL  | NORMZI     |                               | 31 |
| 32 |          |        | ZI +6      |                               | 32 |
| 33 |          | ABVAL  | PUSH       |                               | 33 |
| 34 |          | DSU    | BMN        |                               | 34 |
| 35 |          |        | NORMZI     |                               | 35 |
| 36 |          |        | +3         |                               | 36 |
| 37 |          | DLOAD  | STADR      |                               | 37 |
| 38 |          | STORE  | NORMZI     |                               | 38 |
| 39 |          | VLOAD  | ABVAL      |                               | 39 |
| 40 |          |        | ZI +12D    |                               | 40 |
| 41 |          | PUSH   | DSU        |                               | 41 |
| 42 |          |        | NORMZI     |                               | 42 |
| 43 |          | BMN    | DLOAD      |                               | 43 |
| 44 |          |        | +3         |                               | 44 |
| 45 |          | STADR  |            |                               | 45 |
| 46 |          | STORE  | NORMZI     | # LARGEST ABVAL               | 46 |
| 47 |          | DLOAD  | SXA,1      |                               | 47 |
| 48 |          |        | NORMZI     |                               | 48 |
| 49 |          |        | NORMZI     | # SAVE X1                     | 49 |
| 50 |          | NORM   | INCR,1     |                               | 50 |
| 51 |          |        |            |                               | 51 |
| 52 |          |        |            |                               | 52 |
| 53 |          |        |            |                               | 53 |
| 54 |          |        |            |                               | 54 |
| 55 |          |        |            |                               | 55 |
| 56 |          |        |            |                               | 56 |
| 57 |          |        |            |                               | 57 |
| 58 |          |        |            |                               | 58 |
| 59 |          |        |            |                               | 59 |
| 60 |          |        |            |                               | 60 |

| 142THE |    |
|--------|----|
|        | 1  |
|        | 2  |
|        | 3  |
|        | 4  |
|        | 5  |
|        | 6  |
|        | 7  |
|        | 8  |
|        | 9  |
|        | 10 |
|        | 11 |
|        | 12 |
|        | 13 |
|        | 14 |
|        | 15 |
|        | 16 |
|        | 17 |
|        | 18 |
|        | 19 |
|        | 20 |
|        | 21 |
|        | 22 |
|        | 23 |
|        | 24 |
|        | 25 |
|        | 26 |
|        | 27 |
|        | 28 |
|        | 29 |
|        | 30 |
|        | 31 |
|        | 32 |
|        | 33 |
|        | 34 |
|        | 35 |
|        | 36 |
|        | 37 |
|        | 38 |
|        | 39 |
|        | 40 |
|        | 41 |
|        | 42 |
|        | 43 |
|        | 44 |
|        | 45 |
|        | 46 |
|        | 47 |
|        | 48 |
|        | 49 |
|        | 50 |
|        | 51 |
|        | 52 |
|        | 53 |
|        | 54 |
|        | 55 |
|        | 56 |
|        | 57 |
|        | 58 |
|        | 59 |
|        | 60 |
|        | 61 |
|        | 62 |
|        | 63 |
|        | 64 |
|        | 65 |
|        | 66 |
|        | 67 |
|        | 68 |
|        | 69 |
|        | 70 |
|        | 71 |
|        | 72 |
|        | 73 |
|        | 74 |
|        | 75 |
|        | 76 |
|        | 77 |
|        | 78 |
|        | 79 |
|        | 80 |

# PROGRAM DESCRIPTION -- ENTIRE CONIC SUBROUTINE LOG SECTION      DATE - 1 SEPTEMBER 1967  
# MOD NO. - 0      LOG SECTION - CONIC SUBROUTINES  
# MOD BY KRAUSE      ASSEMBLY - COLOSSUS REVISION 88  
#  
# FUNCTIONAL DESCRIPTION -  
#        THE FOLLOWING SET OF SUBROUTINES SOLVE VARIOUS PROBLEMS INVOLVING THE TRAJECTORY PRODUCED BY A CENTRAL  
# INVERSE-SQUARE FORCE ACTING ON A POINT MASS, AS OUTLINED IN THE CMC AND LGC LUNAR LANDING MISSION GSOP, SECTION  
# 5.5.1.2. A GENERAL USAGE POINT-OF-VIEW WAS TAKEN IN FORMULATING, MECHANIZING, AND SCALING THE SUBROUTINES,  
# RATHER THAN OPTIMIZING EACH FOR A PARTICULAR USE. THEREFORE, MULTIPLE USAGE CAN BE MADE OF THE SUBROUTINES  
# INVOLVING ANY REALISTIC SET OF CONSTRAINTS. IT SHOULD BE NOTED THAT ONLY ONE SET OF CODING IS USED, WHETHER THE  
# EARTH, MOON, OR ANY OTHER CELESTIAL BODY IS SPECIFIED AS THE CENTRAL BODY OF THE PROBLEM, PROVIDED ONE OBSERVES  
# THE INHERENT SCALE CHANGE REQUIRED IN POSITION, VELOCITY, MU, AND TIME, AS OUTLINES IN MISSION PROGRAMMING  
# DEFINITION MEMO NO. 10. THIS CAN BE ACCOMPLISHED BY SIMPLY ADDING TO THE MUTABLE AND INITIALIZING THE SUBROUTINES  
# APPROPRIATELY.  
#  
#        DUE TO THE UNIFORMITY OF THE EQUATIONS INVOLVED, CODING WAS MINIMIZED BY TREATING INDIVIDUAL EQUATIONS AND  
# BLOCKS OF EQUATIONS AS SUBROUTINES OF LOWER RANK WHENEVER POSSIBLE. AS A RESULT, THREE BY-PRODUCTS SUBROUTINES,  
# DIRECTLY USABLE AS INDEPENDENT SUBROUTINES, WERE GENERATED.  
#  
# RESTRICTIONS -  
#        THE ONLY LIMITATION IN THE SCOPE OF THE PROBLEM WHICH CAN BE SOLVED BY A PARTICULAR SUBROUTINE IS THE SCALING  
# LIMIT OF EACH PARAMETER AS SPECIFIED IN THE GSOP. THESE SCALING LIMITS WERE CHOSEN SO THAT ALL FEASIBLE TRAJECTORIES  
# COULD BE HANDLED.  
#  
#        SINCE THE SUBROUTINES (EXCEPT KEPLER) USE COMMON SUBROUTINES OF LOWER RANK WHICH USE ERASABLE OTHER THAN  
# THE PUSHLIST (DUE TO ITS LIMITED SIZE) AND COMMON INTERPRETIVE SWITCHES, THE CONIC SUBROUTINES CANNOT BE ALLOWED  
# TO INTERRUPT EACH OTHER. IT IS UP TO THE USER TO GUARANTEE THIS CONDITION.



# PROGRAM DESCRIPTION - KEPLER SUBROUTINE

DATE - 11 OCTOBER 1967

# MOD NO. -1

LOG SECTION - CONIC SUBROUTINES

# MOD BY KRAUSE

ASSEMBLY - COLOSSUS 103 AND SUNDANCE 222

# MOD NO. - 2 (AUGUST 1968) BY ROBERTSON: TO PERMIT BACKDATING BY MORE THAN ONE ORBITAL PERIOD.

# MOD NO. - 3 (DEC 1968) BY ROBERTSON: SUPPRESSION OF X-MODULO-ING

# MOD NO. - 4 (JAN 1969) BY ROBERTSON: CLEAR OVFIN D AT KEPLER ENTRY

# FUNCTIONAL DESCRIPTION -

# THIS SUBROUTINE, GIVEN AN INITIAL STATE VECTOR AND THE DESIRED TRANSFER TIME THROUGH WHICH THE STATE IS TO  
# BE UPDATED ALONG A CONIC TRAJECTORY, COMPUTES THE NEW, UPDATED STATE VECTOR. THE TRAJECTORY MAY BE ANY CONIC  
# SECTION - CIRCULAR, ELLIPTIC, PARABOLIC, HYPERPOLIC, OR RECTILINEAR WITH RESPECT TO THE EARTH OR THE MOON. THE  
# USE OF THE SUBROUTINE CAN BE EXTENDED USING OTHER PRIMARY BODIES BY SIMPLE ADDITIONS TO THE MUTABLE WITHOUT  
# INTRODUCING ANY CODING CHANGES, ACCEPTING THE INHERENT SCALE FACTOR CHANGES IN POSITION AND VELOCITY. AN ITERATION  
# TECHNIQUE IS UTILIZED IN THE COMPUTATION.

# IF A NEGATIVE TIME-OF-FLIGHT IS INPUT, THE PROGRAM WILL SOLVE FOR THE STATE WHICH WOULD BE PRODUCED BY  
# EXTRAPOLATING THE POSITION BACKWARD IN TIME.

# IF THE ABSOLUTE VALUE OF THE DESIRED TRANSFER TIME EXCEEDS THE ORBITAL PERIOD, THE SUBROUTINE, THROUGH A  
# MODULAR TECHNIQUE, WILL COMPUTE THE STATE CORRESPONDING TO THE DESIRED TIME (WHETHER POSITIVE OR NEGATIVE).

# THE RESTRICTIONS ARE -

- # 1. (PREVIOUS RESTRICTION ON THE NEGATIVE DESIRED TRANSFER TIME IS NOW DELETED.)
- # 2. THE PARAMETERS IN THE PROBLEM CANNOT EXCEED THEIR SCALING LIMITS AS SPECIFIED IN THE GSOP. IF  
# ANY OF THESE LIMITS ARE EXCEEDED, THE RESULTING SOLUTION WILL BE MEANINGLESS.

# THE NUMBER OF ITERATIONS AND, THEREFORE, THE COMPUTATION SPEED IS DEPENDENT ON THE ACCURACY OF THE  
# GUESS, XKFPNEW. THE AGC COMPUTATION TIME IS APPROXIMATELY .061 SECONDS FOR INITIALIZATION, .065 SECONDS FOR THE  
# FINAL COMPUTATIONS, PLUS .083 SECONDS FOR EACH ITERATION.

# REFERENCES -

# R-479, MISSION PROGRAMMING DEFINITION MEMO NO. 10, LUNAR LANDING MISSION GSOP, SECTION 5.5, SGA  
# MEMO 67-4.

# INPUT - ERASABLE INITIALIZATION REQUIRED

| # | #        | * SCALE FACTOR *  | #                                    |
|---|----------|-------------------|--------------------------------------|
| # | VARIABLE | *IN POWERS OF 2 * | DESCRIPTION AND REMARKS              |
| # | -----    | *-----*           | -----                                |
| # | RRECT    | * +29 FOR EARTH * | DP INITIAL POSITION VECTOR IN METERS |
| # |          | * +27 FOR MOON *  |                                      |

```
1 # VRECT * +7 FOR EARTH * DP INITIAL VELOCITY VECTOR IN METERS/CENTISECOND
2 # * +5 FOR MOON *
3 # X1 (38D) * NONE * INDEX REGISTER SET TO -2D OR -10D ACCORDING TO WHETHER THE EARTH OR MOON,
4 # * * * RESPECTIVELY, IS THE CENTRAL BODY
5 # TAU * +28 * DESIRED TRANSFER TIME IN CENTISECONDS (DP)
6 # * * * MAY BE POS OR NEG AND ABSOLUTE VALUE MAY BE GREATER OR LESS THAN ONE ORBITAL
7 # XKEPNEW * +17 FOR EARTH * DP GUESS OF ROOT X OF KEPLERS EQN IN SQRT(METERS). SIGN SHOULD AGREE WITH THAT OF TA
8 # * +16 FOR MOON * AND ABS VALUE SHOULD BE LESS THAN THAT CORRESPONDING TO A PERIOD, VIZ, 2PI S
9 # * * * MAJOR AXIS), FOR SPEED OF CONVERGENCE, BUT IF EITHER CONDITION FAILS, XKEPNE
10 # * * * BY KEPLER TO A POOR BUT VALID GUESS.
11 # TC * +28 * DP PREV. VALUE OF TIME IN CENTISECS. MUST BE LESS THAN ONE ORBITAL PERIOD.
12 # XPREV * +17 FOR EARTH * DP PREV. VALUE OF X IN SQRT(METERS). MUST BE LESS THAN AN X CORRESPONDING TO ONE
13 # * +16 FOR MOON * ORBITAL PERIOD, VIZ, 2PI SQRT(SEMI-MAJOR AXIS)
14 #
15 # SUBROUTINES CALLED -
16 # DELTIME
17 #
18 # CALLING SEQUENCE AND NORMAL EXIT MODES -
19 # KEPRTN-2 GOTO # MUST BE IN INTERPRETIVE MODE BUT OVFINF ARBITRARY.
20 # KEPRTN-1 KEPLER # RETURNS WITH XPREV IN MPAC. PL IS AT 0.
21 # KEPRTN ... # CONTINUE
22 #
23 # KEPLER MUST NOT BE CALLED DIRECTLY SINCE AN INTERRUPTION OF IT WOULD DESTROY THE ERASABLES IT NEEDS TO COMPLETE
24 # THE INTERRUPTED JOB. THEREFORE THE USER MUST CALL CSMCONIC OR LEMCONIC WHICH GUARANTEES NO INTERRUPTS AND WHICH
25 # ALSO CALLS KEPPREP TO COMPUTE A GUESS OF XKEPNEW.
26 #
27 # ABORT EXIT MODES -
28 # NONE
29 #
30 # OUTPUT -
31 #
32 # * SCALE FACTOR *
33 # VARIABLE *IN POWERS OF 2 * DESCRIPTION AND REMARKS
34 # ----- *----- * -----
35 # RCV * +29 FOR EARTH * DP TERMINAL POSITION VECTOR IN METERS
36 # * +27 FOR MOON *
37 # VCV * +7 FOR EARTH * DP TERMINAL VELOCITY VECTOR IN METERS/CENTISEC
38 # * +5 FOR MOON *
39 # TC * +28 * DP TRANSFER TIME IN CENTISECS TO WHICH KEPLER CONVERGED. ALWAYS LESS THAN ONE PERIOD
40 # XPREV * +17 FOR EARTH * DP VALUE OF X IN SQRT(METERS) TO WHICH KEPLER CONVERGED. ALWAYS LESS THAN THE X
41 # * +16 FOR MOON * CORRESPONDING TO ONE PERIOD.
```

# CONIC\_SUBROUTINES

# FOR OTHER OUTPUT WHICH MAY BE OF USE, SEE DEBRIS.

#

# DEBRIS -

#       PARAMETERS WHICH MAY BE OF USE -

#               \* SCALE FACTOR \*

#       VARIABLE       \*IN POWERS OF 2 \*       DESCRIPTION AND REMARKS

#       -----       \*-----\*       -----

#       URRECT       \* +1       \*       DP UNIT VECTOR OF INITIAL POSITION

#       R1       \* +29 FOR EARTH \*       DP MAGNITUDE OF INITIAL POSITION IN METERS

#       ALPHA       \* +27 FOR MOON \*

#       ALPHA       \* -22 FOR EARTH \*       DP INVERSE OF SEMI-MAJOR AXIS IN 1/METERS

#       TMODULO       \* -20 FOR MOON \*

#       TMODULO       \* +28       \*       DP INTEGRAL NUMBER OF PERIODS IN CENTISECS, WHICH WAS SUBTRACTED FROM TAU, TO PRODUC

#       \*       \*       TAU. OF LESS THAN ONE PERIOD.

# PARAMETERS OF NO USE -

#       DP PARAMETERS - EPSILONT, DELX, DELT, RCNORM, XMODULO, PLUS PUSHLIST REGISTERS 0 THROUGH 39D.

## # PROGRAM DESCRIPTION - LAMBERT SUBROUTINE

DATE - 1 SEPTEMBER 1967

# MOD NO. - 0

LOG SECTION - CONIC SUBROUTINES

# MOD BY KRAUSE

ASSEMBLY - COLOSSUS REVISION 88

#

## # FUNCTIONAL DESCRIPTION -

# THIS SUBROUTINE CALCULATES THE INITIAL VELOCITY REQUIRED TO TRANSFER A POINT-MASS ALONG A CONIC TRAJECTORY  
# FROM AN INITIAL POSITION TO A TERMINAL POSITION IN A PRESCRIBED TIME INTERVAL. THE RESULTING TRAJECTORY MAY BE  
# A SECTION OF A CIRCLE, ELLIPSE, PARABOLA, OR HYPERBOLA WITH RESPECT TO THE EARTH OR THE MOON. THE USE OF THE  
# SUBROUTINE CAN BE EXTENDED USING OTHER PRIMARY BODIES BY SIMPLE ADDITIONS TO THE MUTABLE WITHOUT INTRODUCING ANY  
# CODING CHANGES, ACCEPTING THE INHERENT SCALE FACTOR CHANGES IN POSITION AND VELOCITY. AN ITERATION TECHNIQUE IS  
# UTILIZED IN THE COMPUTATION.

#

## # THE RESTRICTIONS ARE: -

- # 1. RECTILINEAR TRAJECTORIES CANNOT BE COMPUTED.
- # 2. AN ACCURACY DEGRADATION OCCURS AS THE COSINE OF THE TRUE ANOMALY DIFFERENCE APPROACHES +1.0.
- # 3. THE ANGLE BETWEEN ANY POSITION VECTOR AND ITS VELOCITY VECTOR MUST BE GREATER THAN 1 DEGREE 47.5 MINUTES  
# AND LESS THAN 178 DEGREES 12.5 MINUTES.
- # 4. NEGATIVE TRANSFER TIME IS AMBIGUOUS AND WILL RESULT IN NO SOLUTION.
- # 5. THE PARAMETERS IN THE PROBLEM MUST NOT EXCEED THEIR SCALING LIMITS SPECIFIED IN THE GSOP. IF THE  
# LIMITS ARE EXCEEDED, THE RESULTING SOLUTION WILL BE MEANINGLESS.

#

# THE NUMBER OF ITERATIONS AND, THEREFORE, THE COMPUTATIONS SPEED IS DEPENDENT ON THE ACCURACY OF THE FIRST  
# GUESS OF THE INDEPENDENT VARIABLE, COGA. THE AGC COMPUTATION TIME IS APPROXIMATELY  
# .105 SECONDS FOR INITIALIZATION, .069 SECONDS FOR FINAL COMPUTATIONS, PLUS .205 SECONDS FOR EACH ITERATION.

#

## # REFERENCES -

# R-479, MISSION PROGRAMMING DEFINITION MEMO NO. 10, LUNAR LANDING MISSION GSOP - SECTION 5.5, SGA MEMO 67-8,  
# SGA MEMO 67-4.

#

## # INPUT - ERASABLE INITIALIZATION REQUIRED

#

| # | # | #        | * SCALE FACTOR *  | # | #                                                                                    |
|---|---|----------|-------------------|---|--------------------------------------------------------------------------------------|
| # | # | VARIABLE | *IN POWERS OF 2 * | # | DESCRIPTION AND REMARKS                                                              |
| # | # | -----    | *-----*           | # | -----                                                                                |
| # | # | R1VEC    | * +29 FOR EARTH * | # | DP INITIAL POSITION VECTOR IN METERS                                                 |
| # | # |          | * +27 FOR MOON *  | # |                                                                                      |
| # | # | R2VEC    | * +29 FOR EARTH * | # | DP TARGET OR TERMINAL POSITION VECTOR IN METERS                                      |
| # | # |          | * +27 FOR MOON *  | # |                                                                                      |
| # | # | TDESIRED | * +28             | # | DP DESIRED TRANSFER TIME IN CENTISECONDS                                             |
| # | # | X1 (38D) | * NONE            | # | INDEX REGISTER SET TO -2D OR -10D ACCORDING TO WHETHER THE EARTH OR MOON,            |
| # | # |          | *                 | # | RESPECTIVELY, IS THE CENTRAL BODY                                                    |
| # | # | GEOMSGN  | * NONE            | # | SP +.5 IF DESIRED TRANSFER ANGLE IS LESS THAN 180 DEGREES, -.5 IF GREATER THAN 180 D |
| # | # | GUESSW   | * NONE            | # | AN INTERPRETER SWITCH TO BE SET IF NO GUESS OF COGA IS AVAILABLE, CLEAR IF A GUESS 0 |

```
1 #
2 # COGA * * COGA IS TO BE USED BY LAMBERT
3 # * +5 * DP GUESS OF COTANGENT OF FLIGHT PATH ANGLE (MEASURED FROM VERTICAL). THIS WILL BE
4 # * * * IGNORED IF GUESSW IS SET.
5 # NORMSW * * AN INTERPRETER SWITCH TO BE SET IF UN IS TO BE AN INPUT TO THE SUBROUTINE, CLEAR IF
6 # * * * LAMBERT IS TO COMPUTE ITS OWN NORMAL (UN).
7 # UN * +1 * DP UNIT NORMAL TO THE DESIRED ORBIT PLANE IN THE DIRECTION OF THE RESULTING ANGULAR
8 # * * * MOMENTUM VECTOR. THIS WILL BE IGNORED IF NORMSW IS CLEAR.
9 # VTARGETAG * * * A S.P. TAG TO BE SET TO ZERO IF LAMBERT IS TO COMPUTE THE VELOCITY AT R2VEC AS WELL
10 # * * * AT R1VEC.
11 # ITERCTR * * * A S.P. COUNTER WHICH SPECIFIES THE MAXIMUM NUMBER OF ITERATIONS ALLOWABLE.
12 # * * * (AN ITERATION MEANS A PASS THRU KEPLER EQN (DELTIME). AT LEAST ONE OF THESE
13 # * * * ALWAYS OCCUR, EVEN IF COGA CORRESPONDING TO SOLUTION WERE INPUT AS A GUESS.)
14 # * * * TWENTY ITERATIONS ARE SUFFICIENT TO SOLVE ALL PROBLEMS INCLUDING THOSE WITHO
15 #
16 # SUBROUTINES CALLED -
17 # GEOM, GETX, DELTIME, ITERATOR, LAMENTER (PART OF NEWSTATE)
18 #
19 # CALLING SEQUENCE AND NORMAL EXIT MODES -
20 # L CALL # MUST BE IN INTERPRETIVE MODE BUT OVFINDD ARBITRARY.
21 # L+1 LAMBERT # RETURNS WITH PL AT 0 AND WITH VVEC IN MPAC IF VTARGETAG WAS NON-ZERO OR VTARGET
22 # # IN MPAC IF VTARGETAG WAS ZERO
23 # L+2 BON # CONTINUE IF SOLNSW CLEAR SINCE SOLUTION IS ACCEPTABLE
24 # L+3 SOLNSW
25 # L+4 LAMABORT
26 #
27 # IF A LAMBERT RESULT IS TO BE A FIRST GUESS FOR THE NEXT LAMBERT CALCULATION, COGA MUST BE PRESERVED AND
28 # GUESSW MUST BE CLEAR FOR EACH SUCCEEDING LAMBERT CALL.
29 #
30 # ABORT EXIT MODES -
31 # IF SOLNSW WAS SET UPON EXITING, EITHER LAMBERT WAS ASKED TO COMPUTE A TRANSFER TOO NEAR 0 OR 360 DEG, OR T
32 # WAS TOO SMALL TO PRODUCE A REALISTIC TRANSFER BETWEEN R1VEC AND R2VEC. IN EITHER CASE THE FIX MUST BE MADE
33 # ACCORDING TO THE NEEDS OF THE PARTICULAR USER. THE ABORT EXIT MODE MAY BE CODED AS ...
34 # LAMBABORT DLOAD ABS # A MEASURE OF THE PROXIMITY TO 0 OR
35 # * 1-CSTH # 360 DEGREES.
36 # DSU BMN
37 # ONEBIT
38 # CHANGER2 # CHANGE R2VEC DIRECTION SLIGHTLY.
39 # DLOAD DAD
40 # TDESIRED
41 # SOMETIME
42 # STCALL TDESIRED # INCREASE TDESIRED
43 # LAMBERT
44 #
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

# OUTPUT -

| # | VARIABLE | * SCALE FACTOR * | DESCRIPTION AND REMARKS                                                              |
|---|----------|------------------|--------------------------------------------------------------------------------------|
| # | -----    | *-----*          | -----                                                                                |
| # | VVEC     | * +7 FOR EARTH * | DP INITIAL VELOCITY VECTOR IN METERS/CENTISECOND REQUIRED TO SATISFY THE BOUNDARY VA |
| # | VTARGET  | * +5 FOR MOON *  | PROBLEM.                                                                             |
| # |          | * +7 FOR EARTH * | DP RESULTANT VELOCITY VECTOR AT R2VEC IN METERS/CENTISECOND.                         |
| # |          | * +5 FOR MOON *  |                                                                                      |
| # | SOLNSW   | * NONE *         | INTERPRETER SWITCH WHICH IS SET IF THE SUBROUTINE CANNOT SOLVE THE PROBLEM, CLEAR IF |
| # |          | *                | SOLUTION EXISTS.                                                                     |

FOR OTHER OUTPUT WHICH MAY BE OF USE, SEE DEBRIS.

# DEBRIS -

PARAMETERS WHICH MAY BE OF USE -

| # | VARIABLE | * SCALE FACTOR *  | DESCRIPTION AND REMARKS                                               |
|---|----------|-------------------|-----------------------------------------------------------------------|
| # | -----    | *-----*           | -----                                                                 |
| # | SNTH     | * +1 *            | DP SIN OF ANGLE BETWEEN R1VEC AND R2VEC                               |
| # | CSTH     | * +1 *            | DP COSINE OF ANGLE                                                    |
| # | 1-CSTH   | * +2 *            | DP 1-CSTH                                                             |
| # | COGA     | * +5 *            | DP COTAN OF INITIAL REQUIRED FLIGHT PATH ANGLE MEASURED FROM VERTICAL |
| # | P        | * +4 *            | DP RATIO OF SEMILATUS RECTUM TO INITIAL RADIUS                        |
| # | R1A      | * +6 *            | DP RATIO OF INITIAL RADIUS TO SEMI-MAJOR AXIS                         |
| # | R1 (32D) | * +29 FOR EARTH * | DP INITIAL RADIUS IN METERS                                           |
| # |          | * +27 FOR MOON *  |                                                                       |
| # | UR1      | * +1 *            | DP UNIT VECTOR OF R1VEC                                               |
| # | U2       | * +1 *            | DP UNIT VECTOR OF R2VEC                                               |

PARAMETERS OF NO USE -

DP PARAMETERS - EPSILONL, CSTH-RHO, TPREV, TERRLAMB, R2, RTNLAMB (SP), PLUS PUSHLIST REGISTER 0 THROUGH 41D  
ADDITIONAL INTERPRETIVE SWITCHES USED - INFINFLG, 360SW, SLOPESW, ORDERSW

# PROGRAM DESCRIPTION - TIME-THETA SUBROUTINE

DATE - 1 SEPTEMBER 1967

# MOD NO. - 0

LOG SECTION - CONIC SUBROUTINES

# MOD BY KRAUSE

ASSEMBLY - COLOSSUS REVISION 88

#

# FUNCTIONAL DESCRIPTION -

# THIS SUBROUTINE, GIVEN AN INITIAL STATE VECTOR AND A DESIRED TRUE-ANOMALY-DIFFERENCE THROUGH WHICH THE  
# STATE IS TO BE UPDATED ALONG A CONIC TRAJECTORY, CALCULATES THE CORRESPONDING TIME-OF-FLIGHT AND, IN ADDITION,  
# PROVIDES THE OPTION OF COMPUTING THE NEW UPDATED STATE VECTOR. THE RESULTING TRAJECTORY MAY BE A SECTION OF A  
# CIRCLE, ELLIPSE, PARABOLA, OR HYPERBOLA WITH RESPECT TO THE EARTH OR THE MOON. THE USE OF THE SUBROUTINE CAN BE  
# EXTENDED USING OTHER PRIMARY BODIES BY SIMPLE ADDITIONS TO THE MUTABLE WITHOUT INTRODUCING ANY CODING CHANGES,  
# ACCEPTING THE INHERENT SCALE FACTOR CHANGES IN POSITION AND VELOCITY.

# THE RESTRICTIONS ARE -

- # 1. THE ANGLE BETWEEN ANY POSITION VECTOR AND ITS VELOCITY VECTOR MUST BE GREATER THAN 1 DEGREE 47.5 MINUTES  
AND LESS THAN 178 DEGREES 12.5 MINUTES.
- # 2. THE PARAMETERS IN THE PROBLEM MUST NOT EXCEED THEIR SCALING LIMITS SPECIFIED IN THE GSOP. IF THE LIMITS  
ARE EXCEEDED, THE RESULTING SOLUTION WILL BE MEANINGLY.

# THE AGC COMPUTATION TIME IS APPROXIMATELY .292 SECONDS.

# REFERENCES -

# R-479, MISSION PROGRAMMING DEFINITION MEMO NO. 10, LUNAR LANDING MISSION GSOP-SECTION 5.5, SGA MEMO 67-8.

# INPUT - ERASABLE INITIALIZATION REQUIRED

| # | VARIABLE | * SCALE FACTOR *  | * IN POWERS OF 2 * | DESCRIPTION AND REMARKS                                                              |
|---|----------|-------------------|--------------------|--------------------------------------------------------------------------------------|
| # | -----    | *-----*           | *                  | -----                                                                                |
| # | RVEC     | * +29 FOR EARTH * | *                  | DP INITIAL POSITION VECTOR IN METERS                                                 |
| # |          | * +27 FOR MOON *  | *                  |                                                                                      |
| # | VVEC     | * +7 FOR EARTH *  | *                  | DP INITIAL VELOCITY VECTOR IN METERS/CENTISECOND                                     |
| # |          | * +5 FOR MOON *   | *                  |                                                                                      |
| # | SNTH     | * +1              | *                  | DP SINE OF TRUE-ANOMALY-DIFFERENCE THROUGH WHICH THE STATE IS TO BE UPDATED          |
| # | CSTH     | * +1              | *                  | DP COSINE OF THE ANGLE                                                               |
| # | RVSX     | * NONE            | *                  | AN INTERPRETIVE SWITCH TO BE SET IF ONLY TIME IS TO BE AN OUTPUT, CLEAR IF THE NEW S |
| # |          | *                 | *                  | IS TO BE COMPUTED ALSO.                                                              |
| # | X1 (38D) | * NONE            | *                  | INDEX REGISTER TO BE SET TO -2D OR -10D ACCORDING TO WHETHER THE EARTH OR MOON,      |
| # |          | *                 | *                  | RESPECTIVELY, IS THE CENTRAL BODY.                                                   |

# SUBROUTINES CALLED -



```
PARAM, GEOM, GETX, DELTIME, NEWSTATE
```

```
CALLING SEQUENCE AND NORMAL EXIT MODES -
```

```
IF ONLY TIME IS DESIRED AS OUTPUT -
```

```
L SET CALL # MUST BE IN INTERPRETIVE MODE BUT OVFLND ARBITRARY.
```

```
L+1 RVSW
```

```
L+2 TIMETHET # RETURN WITH PL AT 0 AND T IN MPAC
```

```
L+3 ... # CONTINUE
```

```
IF THE UPDATE STATE VECTOR IS DESIRED AS WELL -
```

```
L CLEAR CALL # MUST BE IN INTERPRETIVE MODE BUT OVFLND ARBITRARY.
```

```
L+1 RVSW
```

```
L+2 TIMETHET # RETURNS WITH PL AT 6. THE INITIAL POSITION VECTOR IS IN OD OF THE PUSHLIST AND
```

```
THE INITIAL VELOCITY VECTOR IN MPAC.
```

```
L+3 STOVL NEWVVEC
```

```
L+4 STADR
```

```
L+5 STORE NEWRVEC # NEWVVEC AND NEWRVEC ARE SYMBOLIC REPRESENTATIONS OF THE USERS LOCATIONS.
```

```
L+6 ... # CONTINUE.
```

```
ABORT EXIT MODES -
```

```
IF COGAFLEG AND/OR INFINFLG IS SET AT THE EXIT TO TIME-THETA, TIME-THETA WILL TRANSFER TO POODOO WITH
```

```
AN ALARM CODE (ORIGINALLY 00607), AND NOT RETURN TO THE CALLING PROGRAM. (PCR 692 AND 721).
```

```
OUTPUT -
```

```
* SCALE FACTOR *
```

```
VARIABLE *IN POWERS OF 2 * DESCRIPTION AND REMARKS
```

```
----- *-----*
```

```
T (30D) * +28 * DP TRANSFER TIME IN CENTISECONDS
```

```
INFINFLG * NONE * AN INTERPRETIVE SWITCH WHICH IS SET IF THE TRANSFER ANGLE REQUIRES CLOSURE THROUGH
```

```
INFINITY (NO SOLUTION), CLEAR IF A PHYSICAL SOLUTION IS POSSIBLE.
```

```
COGAFLEG * NONE * AN INTERPRETIVE SWITCH WHICH IS SET IF RESTRICTION 1 HAS BEEN VIOLATED (NO SOLUTION)
```

```
* * CLEAR IF NOT.
```

```
IN ADDITION, IF RVSW IS CLEAR, THE FOLLOWING ARE OUTPUT -
```

```
MPAC - * +7 FOR EARTH * DP TERMINAL VELOCITY VECTOR IN METERS/CENTISEC.
```

```
MPAC +5 * +5 FOR MOON *
```

```
OD - 5D * +29 FOR EARTH * DP TERMINAL POSITION VECTOR IN METERS (PL AT 6D)
```

```
* +27 FOR MOON *
```

```
FOR OTHER OUTPUT WHICH MAY BE OF USE, SEE DEBRIS.
```



# DEBRIS -

#       PARAMETERS WHICH MAY BE OF USE -

| # | VARIABLE | * SCALE FACTOR *  | *IN POWERS OF 2 * | DESCRIPTION AND REMARKS                                             |
|---|----------|-------------------|-------------------|---------------------------------------------------------------------|
| # | -----    | *-----*           |                   | -----                                                               |
| # | R1 (32D) | * +29 FOR EARTH * |                   | DP MAGNITUDE OF INITIAL POSITION VECTOR, RVEC, IN METERS            |
| # |          | * +27 FOR MOON *  |                   |                                                                     |
| # | R1A      | * +6              | *                 | DP RATIO OF R1 TO SEMIMAJOR AXIS (NEG. FOR HYPERBOLIC TRAJECTORIES) |
| # | P        | * +4              | *                 | DP RATIO OF SEMILATUS RECTUM TO R1                                  |
| # | COGA     | * +5              | *                 | DP COTAN OF ANGLE BETWEEN RVEC AND VVEC                             |
| # | UR1      | * +1              | *                 | DP UNIT VECTOR OF RVEC                                              |
| # | U2       | * +1              | *                 | DP UNIT VECTOR OF VVEC                                              |
| # | UN       | * +1              | *                 | DP UNIT VECTOR OF UR1*U2                                            |

#       PARAMETERS OF NO USE -

#       SP PARAMETERS -- RTNTT, GEOMSGN, RTNPRM, MAGVEC2=R2 (DP), PLUS PUSHLIST LOCATIONS 0-11D, 14D-21D, 24D-39D, 41D

#       ADDITIONAL INTERPRETIVE SWITCHES USED -- NORMSW, 360SW

# PROGRAM DESCRIPTION - TIME-RADIUS SUBROUTINE

DATE - 11 OCTOBER 1967

# MOD NO. -1

LOG SECTION - CONIC SUBROUTINES

# MOD BY KRAUSE

ASSEMBLY - COLOSSUS REVISION 88

# FUNCTIONAL DESCRIPTION -

# THIS SUBROUTINE, GIVEN AN INITIAL STATE VECTOR AND A DESIRED RADIUS TO WHICH THE STATE IS TO BE UPDATED ALONG A CONIC TRAJECTORY, CALCULATES THE CORRESPONDING TIME-OF-FLIGHT AND, IN ADDITION, PROVIDES THE OPTION OF COMPUTING THE NEW UPDATED STATE VECTOR. THE RESULTING TRAJECTORY MAY BE A SECTION OF A CIRCLE, ELLIPSE, PARABOLA, OR HYPERBOLA WITH RESPECT TO THE EARTH OR THE MOON. THE USE OF THE SUBROUTINE CAN BE EXTENDED USING OTHER PRIMARY BODIES BY SIMMPE ADDITIONS TO THE MUTABLE WITHOUT INTRODUCING ANY CODING CHANGES, ACCEPTING THE INHERENT SCALE FACTOR CHANGES IN POSITION AND VELOCITY.

# IF THE DESIRED RADIUS IS BEYOND THE RADIUS OF APOCENTER OF THE CONIC OR BELOW THE RADIUS OF PERICENTER, APSESW WILL BE SET AND THE SUBROUTINE WILL RETURN THE APOCENTER OR PERICENTER SOLUTION, RESPECTIVELY.

# THE RESTRICTIONS ARE -

- # 1. THE ANGLE BETWEEN ANY POSITION VECTOR AND ITS VELOCITY VECTOR MUST BE GREATER THAN 1 DEGREE 47.5 MINUTES AND LESS THAN 178 DEGREES 12.5 MINUTES.
- # 2. THE PARAMETERS IN THE PROBLEM MUST NOT EXCEED THEIR SCALING LIMITS SPECIFIED IN THE GSOP. IF THE LIMITS ARE EXCEEDED, THE RESULTING SOLUTION WILL BE MEANINGLESS.
- # 3. AN ACCURACY DEGRADATION OCCURS AS THE SENSITIVITIES OF TIME AND UPDATED STATE VECTOR TO CHANGES IN RDESIRED INCREASE. THIS WILL OCCUR NEAR EITHER APSIS OF THE CONIC AND WHEN THE CONIC IS NEARLY CIRCULAR. IN PARTICULAR, IF THE CONIC IS AN EXACT CIRCLE, THE PROBLEM IS UNDEFINED AND THE SUBROUTINE WILL ABORT.

# THE AGC COMPUTATION TIME IS APPROXIMATELY .363 SECONDS.

# REFERENCES -

# R-479, MISSION PROGRAMMING DEFINITION MEMO NO. 10, LUNAR LANDING MISSION GSOP-SECTION 5.5, SGA MEMO 67-8.

# INPUT - ERASABLE INITIALIZATION REQUIRED.

| # | VARIABLE | * SCALE FACTOR *  | DESCRIPTION AND REMARKS                                                                                                                                        |
|---|----------|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| # | -----    | *-----*           | -----                                                                                                                                                          |
| # | RVEC     | * +29 FOR EARTH * | DP INITIAL POSITION VECTOR IN METERS                                                                                                                           |
| # |          | * +27 FOR MOON *  |                                                                                                                                                                |
| # | VVEC     | * +7 FOR EARTH *  | DP INITIAL VELOCITY VECTOR IN METERS/CENTISECOND                                                                                                               |
| # |          | * +5 FOR MOON *   |                                                                                                                                                                |
| # | RDESIRED | * +29 FOR EARTH * | DP TERMINAL RADIAL DISTANCE ON CONIC TRAJECTORY FOR WHICH TRANSFER TIME IS TO BE COMPUTED                                                                      |
| # |          | * +27 FOR MOON *  |                                                                                                                                                                |
| # | SGNRDOT  | * NONE *          | SP TAG SET TO +.5 OR -.5 ACCORDING TO WHETHER THE RADIAL VELOCITY AT RDESIRED IS TO POSITIVE OR NEGATIVE, RESPECTIVELY. THIS TAG REDUCES THE DOUBLE-VALUED PRO |
| # |          | * *               |                                                                                                                                                                |

```

* * SINGLE-VALUED PROBLEM.
X1 (38D) * NONE * INDEX REGISTER TO BE SET TO -2D OR -10D ACCORDING TO WHETHER THE EARTH OR MOON,
RVSW * NONE * RESPECTIVELY, IS THE CENTRAL BODY.
* * AN INTERPRETIVE SWITCH TO BE SET IF ONLY TIME IS TO BE AN OUTPUT, CLEAR IF THE NEW S
* * IS TO BE COMPUTED ALSO.

```

```

SUBROUTINES CALLED -
PARAM, GEOM, GETX, DELTIME, NEWSTATE

```

```

CALLING SEQUENCE AND NORMAL EXIT MODES -
#

```

```

IF ONLY TIME IS DESIRED AS OUTPUT -
L SET CALL # MUST BE IN INTERPRETIVE MODE BUT OVFLND ARBITRARY.
L+1 RVSW
L+2 TIMERAD # RETURN WITH PL AT 0 AND T IN MPAC
L+3 ... # CONTINUE

```

```

IF THE UPDATE STATE VECTOR IS DESIRED AS WELL -
L CLEAR CALL # MUST BE IN INTERPRETIVE MODE BUT OVFLND ARBITRARY.
L+1 RVSW
L+2 TIMERAD # RETURNS WITH PL AT 6. THE INITIAL POSITION VECTOR IS IN OD OF THE PUSHLIST AND
L+3 STOVL NEWVVEC # THE INITIAL VELOCITY VECTOR IN MPAC.
L+4 STADR
L+5 STORE NEWRVEC # NEWVVEC AND NEWRVEC ARE SYMBOLIC REPRESENTATIONS OF THE USERS LOCATIONS.
...

```

```

ABORT EXIT MODES -
IF SOLNSW AND/OR COGAFLAG AND/OR INFINFLG IS SET AT THE EXIT TO TIME-RADIUS, TIME-RADIUS WILL TRANSFER
TO POODOO WITH AN ALARM CODE (ORIGINALLY 00607), AND NOT RETURN TO THE CALLING PROGRAM. (PCR 692 & 721)

```

```

OUTPUT -

```

| # | VARIABLE | * SCALE FACTOR * | * DESCRIPTION AND REMARKS                                                                                                                             |
|---|----------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| # | -----    | *-----*          | -----                                                                                                                                                 |
| # | T (30D)  | * +28            | DP TRANSFER TIME IN CENTISECONDS.                                                                                                                     |
| # | INFINFLG | * NONE           | AN INTERPRETIVE SWITCH WHICH IS SET IF RDESIRED AND SGNRDOT REQUIRE CLOSURE THROUGH INFINITY (NO SOLUTION), CLEAR IF A PHYSICAL SOLUTION IS POSSIBLE. |
| # | COGAFLAG | * NONE           | AN INTERPRETIVE SWITCH WHICH IS SET IF RESTRICTION 1 HAS BEEN VIOLATED (NO SOLUTION) CLEAR IF NOT.                                                    |
| # | APSESW   | * NONE           | AN INTERPRETIVE SWITCH WHICH IS SET IF RDESIRED WAS GREATER THAN RADIUS OF APOCENTER                                                                  |

```
IN ADDITION, IF RVSW IS CLEAR, THE FOLLOWING ARE OUTPUT -
```

```
MPAC - * +7 FOR EARTH * DP TERMINAL VELOCITY VECTOR IN METERS/CENTISEC.
```

# MPAC +5 \* +5 FOR MOON \*

[illegible]

```
* +27 FOR MOON *
```

```
#
FOR OTHER OUTPUT WHICH MAY BE OF USE, SEE DEBRIS.
```

# DEBRIS -

# PARAMETERS WHICH MAY BE OF USE -

```
* SCALE FACTOR *
```

| # | VARIABLE | *IN POWERS OF 2 * | DESCRIPTION AND REMARKS |
|---|----------|-------------------|-------------------------|
|---|----------|-------------------|-------------------------|

# ----- \*-----\*

```
R1 (32D) * +29 FOR EARTH * DP MAGNITUDE OF INITIAL POSITION VECTOR, RVEC, IN METERS
```

|   |                  |
|---|------------------|
| # | * +27 FOR MOON * |
|---|------------------|

| # | R1A | * +6 | * | DP RATIO OF R1 TO SEMIMAJOR AXIS (NEG. FOR HYPERBOLIC TRAJECTORIES) |
|---|-----|------|---|---------------------------------------------------------------------|
|---|-----|------|---|---------------------------------------------------------------------|

| # | P | * +4 | * | DP RATIO OF SEMILATUS RECTUM TO RL |
|---|---|------|---|------------------------------------|
|---|---|------|---|------------------------------------|

| #  | COGA   | * +5   | *      | DP COTAN OF ANGLE BETWEEN RVEC AND VVEC |
|----|--------|--------|--------|-----------------------------------------|
| 1  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 2  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 3  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 4  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 5  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 6  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 7  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 8  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 9  | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 10 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 11 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 12 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 13 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 14 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 15 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 16 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 17 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 18 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 19 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 20 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 21 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 22 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 23 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 24 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 25 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 26 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 27 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 28 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 29 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 30 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 31 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 32 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 33 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 34 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 35 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 36 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 37 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 38 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 39 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 40 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 41 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 42 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 43 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 44 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 45 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 46 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 47 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 48 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 49 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 50 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 51 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 52 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 53 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 54 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 55 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 56 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 57 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 58 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 59 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 60 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 61 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 62 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 63 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 64 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 65 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 66 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 67 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 68 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 69 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 70 | 0.9876 | 0.9876 | 0.9876 | 0.9876                                  |
| 71 | 0.9876 | 0.9876 | 0.9876 | 0.9876</                                |

```
UR1 * +1 * DP UNIT VECTOR OF RVEC
```

```
U2 * +1 * DP UNIT VECTOR OF VVEC
```

| #  | UN | * +1 | * | DP UNIT VECTOR OF UR1*U2 |
|----|----|------|---|--------------------------|
| 1  | 1  | 1    | 1 | 1                        |
| 2  | 1  | 1    | 1 | 1                        |
| 3  | 1  | 1    | 1 | 1                        |
| 4  | 1  | 1    | 1 | 1                        |
| 5  | 1  | 1    | 1 | 1                        |
| 6  | 1  | 1    | 1 | 1                        |
| 7  | 1  | 1    | 1 | 1                        |
| 8  | 1  | 1    | 1 | 1                        |
| 9  | 1  | 1    | 1 | 1                        |
| 10 | 1  | 1    | 1 | 1                        |
| 11 | 1  | 1    | 1 | 1                        |
| 12 | 1  | 1    | 1 | 1                        |
| 13 | 1  | 1    | 1 | 1                        |
| 14 | 1  | 1    | 1 | 1                        |
| 15 | 1  | 1    | 1 | 1                        |
| 16 | 1  | 1    | 1 | 1                        |
| 17 | 1  | 1    | 1 | 1                        |
| 18 | 1  | 1    | 1 | 1                        |
| 19 | 1  | 1    | 1 | 1                        |
| 20 | 1  | 1    | 1 | 1                        |
| 21 | 1  | 1    | 1 | 1                        |
| 22 | 1  | 1    | 1 | 1                        |
| 23 | 1  | 1    | 1 | 1                        |
| 24 | 1  | 1    | 1 | 1                        |
| 25 | 1  | 1    | 1 | 1                        |
| 26 | 1  | 1    | 1 | 1                        |
| 27 | 1  | 1    | 1 | 1                        |
| 28 | 1  | 1    | 1 | 1                        |
| 29 | 1  | 1    | 1 | 1                        |
| 30 | 1  | 1    | 1 | 1                        |
| 31 | 1  | 1    | 1 | 1                        |
| 32 | 1  | 1    | 1 | 1                        |
| 33 | 1  | 1    | 1 | 1                        |
| 34 | 1  | 1    | 1 | 1                        |
| 35 | 1  | 1    | 1 | 1                        |
| 36 | 1  | 1    | 1 | 1                        |
| 37 | 1  | 1    | 1 | 1                        |
| 38 | 1  | 1    | 1 | 1                        |
| 39 | 1  | 1    | 1 | 1                        |
| 40 | 1  | 1    | 1 | 1                        |
| 41 | 1  | 1    | 1 | 1                        |
| 42 | 1  | 1    | 1 | 1                        |
| 43 | 1  | 1    | 1 | 1                        |
| 44 | 1  | 1    | 1 | 1                        |
| 45 | 1  | 1    | 1 | 1                        |
| 46 | 1  | 1    | 1 | 1                        |
| 47 | 1  | 1    | 1 | 1                        |
| 48 | 1  | 1    | 1 | 1                        |
| 49 | 1  | 1    | 1 | 1                        |
| 50 | 1  | 1    | 1 | 1                        |
| 51 | 1  | 1    | 1 | 1                        |
| 52 | 1  | 1    | 1 | 1                        |
| 53 | 1  | 1    | 1 | 1                        |
| 54 | 1  | 1    | 1 | 1                        |
| 55 | 1  | 1    | 1 | 1                        |
| 56 | 1  | 1    | 1 | 1                        |
| 57 | 1  | 1    | 1 | 1                        |
| 58 | 1  | 1    | 1 | 1                        |
| 59 | 1  | 1    | 1 | 1                        |
| 60 | 1  | 1    | 1 | 1                        |
| 61 | 1  | 1    | 1 | 1                        |
| 62 | 1  | 1    | 1 | 1                        |
| 63 | 1  | 1    | 1 | 1                        |
| 64 | 1  | 1    | 1 | 1                        |
| 65 | 1  | 1    | 1 | 1                        |
| 66 | 1  | 1    | 1 | 1                        |
| 67 | 1  | 1    | 1 | 1                        |
| 68 | 1  | 1    | 1 | 1                        |
| 69 | 1  | 1    | 1 | 1                        |
| 70 | 1  | 1    | 1 | 1                        |
| 71 | 1  | 1    | 1 | 1                        |
| 72 | 1  | 1    | 1 | 1                        |
| 73 | 1  | 1    | 1 | 1                        |
| 74 | 1  | 1    | 1 | 1                        |
| 75 | 1  | 1    | 1 | 1                        |
| 76 | 1  | 1    | 1 | 1                        |
| 77 | 1  | 1    | 1 | 1                        |
| 78 | 1  | 1    | 1 | 1                        |
| 79 | 1  | 1    | 1 | 1                        |
| 80 | 1  | 1    | 1 | 1                        |
| 81 | 1  | 1    | 1 | 1                        |
| 82 | 1  | 1    | 1 | 1                        |
| 83 | 1  | 1    | 1 | 1                        |
| 84 | 1  | 1    | 1 | 1                        |
| 85 | 1  | 1    | 1 | 1                        |
| 86 | 1  | 1    | 1 | 1                        |
| 87 | 1  | 1    | 1 | 1                        |
| 88 | 1  | 1    | 1 | 1                        |
| 89 | 1  | 1    | 1 | 1                        |
| 90 | 1  | 1    | 1 | 1                        |
| 91 | 1  | 1    | 1 | 1                        |
| 92 | 1  | 1    | 1 | 1                        |
| 93 | 1  | 1    | 1 | 1                        |
| 94 | 1  | 1    | 1 | 1                        |

| #  | CSTH | * +1 | * | DP | COSINE OF TRUE ANOMALY DIFFERENCE BETWEEN RVEC AND RDESIRED. |
|----|------|------|---|----|--------------------------------------------------------------|
| 1  | 1    | 1    | 1 | 1  | 1                                                            |
| 2  | 1    | 1    | 1 | 1  | 1                                                            |
| 3  | 1    | 1    | 1 | 1  | 1                                                            |
| 4  | 1    | 1    | 1 | 1  | 1                                                            |
| 5  | 1    | 1    | 1 | 1  | 1                                                            |
| 6  | 1    | 1    | 1 | 1  | 1                                                            |
| 7  | 1    | 1    | 1 | 1  | 1                                                            |
| 8  | 1    | 1    | 1 | 1  | 1                                                            |
| 9  | 1    | 1    | 1 | 1  | 1                                                            |
| 10 | 1    | 1    | 1 | 1  | 1                                                            |
| 11 | 1    | 1    | 1 | 1  | 1                                                            |
| 12 | 1    | 1    | 1 | 1  | 1                                                            |
| 13 | 1    | 1    | 1 | 1  | 1                                                            |
| 14 | 1    | 1    | 1 | 1  | 1                                                            |
| 15 | 1    | 1    | 1 | 1  | 1                                                            |
| 16 | 1    | 1    | 1 | 1  | 1                                                            |
| 17 | 1    | 1    | 1 | 1  | 1                                                            |
| 18 | 1    | 1    | 1 | 1  | 1                                                            |
| 19 | 1    | 1    | 1 | 1  | 1                                                            |
| 20 | 1    | 1    | 1 | 1  | 1                                                            |
| 21 | 1    | 1    | 1 | 1  | 1                                                            |
| 22 | 1    | 1    | 1 | 1  | 1                                                            |
| 23 | 1    | 1    | 1 | 1  | 1                                                            |
| 24 | 1    | 1    | 1 | 1  | 1                                                            |
| 25 | 1    | 1    | 1 | 1  | 1                                                            |
| 26 | 1    | 1    | 1 | 1  | 1                                                            |
| 27 | 1    | 1    | 1 | 1  | 1                                                            |
| 28 | 1    | 1    | 1 | 1  | 1                                                            |
| 29 | 1    | 1    | 1 | 1  | 1                                                            |
| 30 | 1    | 1    | 1 | 1  | 1                                                            |
| 31 | 1    | 1    | 1 | 1  | 1                                                            |
| 32 | 1    | 1    | 1 | 1  | 1                                                            |
| 33 | 1    | 1    | 1 | 1  | 1                                                            |
| 34 | 1    | 1    | 1 | 1  | 1                                                            |
| 35 | 1    | 1    | 1 | 1  | 1                                                            |
| 36 | 1    | 1    | 1 | 1  | 1                                                            |
| 37 | 1    | 1    | 1 | 1  | 1                                                            |
| 38 | 1    | 1    | 1 | 1  | 1                                                            |
| 39 | 1    | 1    | 1 | 1  | 1                                                            |
| 40 | 1    | 1    | 1 | 1  | 1                                                            |
| 41 | 1    | 1    | 1 | 1  | 1                                                            |
| 42 | 1    | 1    | 1 | 1  | 1                                                            |
| 43 | 1    | 1    | 1 | 1  | 1                                                            |
| 44 | 1    | 1    | 1 | 1  | 1                                                            |
| 45 | 1    | 1    | 1 | 1  | 1                                                            |
| 46 | 1    | 1    | 1 | 1  | 1                                                            |
| 47 | 1    | 1    | 1 | 1  | 1                                                            |
| 48 | 1    | 1    | 1 | 1  | 1                                                            |
| 49 | 1    | 1    | 1 | 1  | 1                                                            |
| 50 | 1    | 1    | 1 | 1  | 1                                                            |
| 51 | 1    | 1    | 1 | 1  | 1                                                            |
| 52 | 1    | 1    | 1 | 1  | 1                                                            |
| 53 | 1    | 1    | 1 | 1  | 1                                                            |
| 54 | 1    | 1    | 1 | 1  | 1                                                            |
| 55 | 1    | 1    | 1 | 1  | 1                                                            |
| 56 | 1    | 1    | 1 | 1  | 1                                                            |
| 57 | 1    | 1    | 1 | 1  | 1                                                            |
| 58 | 1    | 1    | 1 | 1  | 1                                                            |
| 59 | 1    | 1    | 1 | 1  | 1                                                            |
| 60 | 1    | 1    | 1 | 1  | 1                                                            |
| 61 | 1    | 1    | 1 | 1  | 1                                                            |
| 62 | 1    | 1    | 1 | 1  | 1                                                            |
| 63 | 1    | 1    | 1 | 1  | 1                                                            |
| 64 | 1    | 1    | 1 | 1  | 1                                                            |
| 65 | 1    | 1    | 1 | 1  | 1                                                            |
| 66 | 1    | 1    | 1 | 1  | 1                                                            |
| 67 | 1    | 1    | 1 | 1  | 1                                                            |
| 68 | 1    | 1    | 1 | 1  | 1                                                            |
| 69 | 1    | 1    | 1 | 1  | 1                                                            |
| 70 | 1    | 1    | 1 | 1  | 1                                                            |
| 71 | 1    | 1    | 1 | 1  | 1                                                            |
| 72 | 1    | 1    | 1 | 1  | 1                                                            |
| 73 | 1    | 1    | 1 | 1  | 1                                                            |
| 74 | 1    | 1    | 1 | 1  | 1                                                            |
| 75 | 1    | 1    | 1 | 1  | 1                                                            |
| 76 | 1    | 1    | 1 | 1  | 1                                                            |
| 77 | 1    | 1    | 1 | 1  | 1                                                            |
| 78 | 1    | 1    | 1 | 1  | 1                                                            |
| 79 | 1    | 1    | 1 | 1  | 1                                                            |
| 80 | 1    | 1    | 1 | 1  | 1                                                            |
| 81 | 1    | 1    |   |    |                                                              |

| # | SNTH | * +1 | * | DP SINE OF TRUE ANOMALY DIFFERENCE. |
|---|------|------|---|-------------------------------------|
|---|------|------|---|-------------------------------------|

# PARAMETERS OF NO USE -

```
SP PARAMETERS -- RTNTT, GEOMSGN, RTNPRM, MAGVEC2=R2 (DP), PLUS PUSHLIST LOCATIONS 0-11D, 14D-21D, 24D-39D, 41D
```

```
#
ADDITIONAL INTERPRETIVE SWITCHES USED -- NORMSW, 360SW
```

# PROGRAM DESCRIPTION - APSIDES SUBROUTINE

DATE - 1 SEPTEMBER 1967

# MOD NO. - 0

LOG SECTION - CONIC SUBROUTINES

# MOD BY KRAUSE

ASSEMBLY - COLOSSUS REVISION 88

# FUNCTIONAL DESCRIPTION -

# THIS SUBROUTINE, GIVEN AN INITIAL STATE VECTOR CALCULATES THE RADIUS OF PERICENTER AND OF APOCENTER AND THE ECCENTRICITY OF THE RESULTING CONIC TRAJECTORY, WHICH MAY BE A STRAIGHT LINE, CIRCLE, ELLIPSE, PARABOLA, OR HYPERBOLA WITH RESPECT TO THE EARTH OR THE MOON. THE USE OF THE SUBROUTINE CAN BE EXTENDED USING OTHER PRIMARY BODIES BY SIMPLE ADDITIONS TO THE MUTABLE WITHOUT INTRODUCING ANY CODING CHANGES, ACCEPTING THE INHERENT SCALE FACTOR CHANGES IN POSITION AND VELOCITY.

# THE RESTRICTIONS ARE -

- # 1. IF APOCENTER IS BEYOND THE SCALING OF POSITION, THE SCALE FACTOR LIMIT (536,870,910 METERS WITH RESPECT TO THE EARTH OR 134,217,727.5 METERS WITH RESPECT TO THE MOON) WILL BE RETURNED.
- # 2. THE PARAMETERS IN THE PROBLEM MUST NOT EXCEED THEIR SCALING LIMITS SPECIFIED IN THE GSOP. IF THE LIMITS ARE EXCEEDED, THE RESULTING SOLUTION WILL BE MEANINGLESS.

# THE AGC COMPUTATION TIME IS APPROXIMATELY .103 SECONDS.

# REFERENCES -

# MISSION PROGRAMMING DEFINITION MEMO NO. 10, LUNAR LANDING MISSION GSOP-SECTION 5.5

# INPUT - ERASABLE INITIALIZATION REQUIRED

| # | VARIABLE | * SCALE FACTOR *  | *IN POWERS OF 2 * | DESCRIPTION AND REMARKS                                                                                            |
|---|----------|-------------------|-------------------|--------------------------------------------------------------------------------------------------------------------|
| # | -----    | *-----*           |                   | -----                                                                                                              |
| # | RVEC     | * +29 FOR EARTH * |                   | DP INITIAL POSITION VECTOR IN METERS                                                                               |
| # |          | * +27 FOR MOON *  |                   |                                                                                                                    |
| # | VVEC     | * +7 FOR EARTH *  |                   | DP INITIAL VELOCITY VECTOR IN METERS/CENTISECOND                                                                   |
| # |          | * +5 FOR MOON *   |                   |                                                                                                                    |
| # | X1 (38D) | * NONE            | *                 | INDEX REGISTER TO BE SET TO -2D OR -10D ACCORDING TO WHETHER THE EARTH OR MOON, RESPECTIVELY, IS THE CENTRAL BODY. |
| # |          | *                 | *                 |                                                                                                                    |

# SUBROUTINES CALLED -

# PARAM, GEOM

# CALLING SEQUENCE AND NORMAL EXIT MODES -

```
1 # IF ONLY TIME IS DESIRED AS OUTPUT -
2 # L CALL # MUST BE IN INTERPRETIVE MODE BUT OVFINDD ARBITRARY.
3
4 # L+1 APSIDES # RETURNS WITH PL AT 0, RADIUS OF APOCENTER IN MPAC AND RADIUS OF PERICENTER IN OD
5 # L+2 STODL APOAPSE
6 # L+3 OD
7 # L+4 STORE PERIAPSE # APOAPSE AND PERIAPSE ARE SYMBOLIC REPRESENTATIONS OF THE USERS LOCATIONS
8 # L+5 ... # CONTINUE
9
10 # OUTPUT -
11 # * SCALE FACTOR *
12 # VARIABLE *IN POWERS OF 2 * DESCRIPTION AND REMARKS
13 # ----- *-----*
14 # MPAC * +29 FOR EARTH * DP RADIUS OF APOCENTER IN METERS
15 # * +27 FOR MOON *
16 # OD-1D * +29 FOR EARTH * DP RADIUS OF PERICENTER IN METERS
17 # * +27 FOR MOON *
18 # ECC * +3 * DP ECCENTRICITY OF CONIC TRAJECTORY.
19
20 # FOR OTHER OUTPUT WHICH MAY BE OF USE, SEE DEBRIS.
21
22 # DEBRIS -
23 #
24 # PARAMETERS WHICH MAY BE OF USE -
25 #
26 # * SCALE FACTOR *
27 # VARIABLE *IN POWERS OF 2 * DESCRIPTION AND REMARKS
28 # ----- *-----*
29 # R1 (32D) * +29 FOR EARTH * DP MAGNITUDE OF INITIAL POSITION VECTOR, RVEC, IN METERS
30 # * +27 FOR MOON *
31 # R1A * +6 * DP RATIO OF R1 TO SEMI-MAJOR AXIS (NEG. FOR HYPERBOLIC TRAJECTORIES)
32 # P * +4 * DP RATIO OF SEMILATUS RECTUM TO R1
33 # COGA * +5 * DP COTAN OF ANGLE BETWEEN RVEC AND VVEC
34 # UR1 * +1 * DP UNIT VECTOR OF RVEC
35 # U2 * +1 * DP UNIT VECTOR OF VVEC
36 # UN * +1 * DP UNIT VECTOR OF UR1*U2
37 # MAGVEC2 * +7 FOR EARTH * DP MAGNITUDE OF VVEC
38 # * +5 FOR MOON *
39 #
40 # PARAMETERS OF NO USE -
41 # SP PARAMETERS - RTNAPSE, GEOMSGN, RTNPRM, PLUS PUSHLIST LOCATIONS 0-5, 10D-11D, 14D-21D, 31D-38D.
42 # ADDITIONAL INTERPRETIVE SWITCHES USED - NORMSW
43
44 SETLOC CONICS
```

|    |         |                              |            |    |
|----|---------|------------------------------|------------|----|
| 1  |         |                              |            | 1  |
| 2  | BANK    |                              |            | 2  |
| 3  |         |                              |            | 3  |
| 4  | KEPLERN | COUNT*                       | \$\$/CONIC | 4  |
| 5  |         | EBANK=                       | UR1        | 5  |
| 6  |         | SETPD                        | BOV        | 6  |
| 7  | 0       |                              |            | 7  |
| 8  | +1      |                              |            | 8  |
| 9  | VLOAD*  |                              |            | 9  |
| 10 | STOVL   | MUTABLE,1                    |            | 10 |
| 11 |         | 14D                          |            | 11 |
| 12 |         | RRECT                        |            | 12 |
| 13 | UNIT    | SSP                          |            | 13 |
| 14 |         | ITERCTR                      |            | 14 |
| 15 |         | 20D                          |            | 15 |
| 16 | STODL   | URRECT                       |            | 16 |
| 17 |         | 36D                          |            | 17 |
| 18 |         | R1                           |            | 18 |
| 19 | DOT     | RRECT                        |            | 19 |
| 20 |         | SL1R                         |            | 20 |
| 21 |         | VRECT                        |            | 21 |
| 22 | DMP     | SL1R                         |            | 22 |
| 23 |         | 1/ROOTMU                     |            | 23 |
| 24 |         | 1/ROOTMU                     |            | 24 |
| 25 | VSQ     | VRECT                        |            | 25 |
| 26 |         | DMPR                         |            | 26 |
| 27 |         | 1/MU                         |            | 27 |
| 28 | DMP     | SL3                          |            | 28 |
| 29 |         | R1                           |            | 29 |
| 30 |         | ROUND                        |            | 30 |
| 31 | STORE   | D1/64                        |            | 31 |
| 32 |         | KEPC2                        |            | 32 |
| 33 |         | C2=RV.V/MU -1 (+6)           |            | 33 |
| 34 | BDSU    | SR1R                         |            | 34 |
| 35 |         | D1/64                        |            | 35 |
| 36 |         |                              |            | 36 |
| 37 | STORE   | R1                           |            | 37 |
| 38 |         | ALPHA                        |            | 38 |
| 39 |         | ALPHA=(1-C2)/R1 (-22 OR -20) |            | 39 |
| 40 | BPL     | DLOAD                        |            | 40 |
| 41 |         | 1REV                         |            | 41 |
| 42 |         | -50SC                        |            | 42 |
| 43 | DDV     | BOV                          |            | 43 |
| 44 |         | ALPHA                        |            | 44 |
| 45 |         | STOREMAX                     |            | 45 |
| 46 | SQRT    | GOTO                         |            | 46 |
| 47 |         | STOREMAX                     |            | 47 |
| 48 |         |                              |            | 48 |
| 49 | 1REV    | SQRT                         | BDDV       | 49 |
| 50 |         |                              |            | 50 |
| 51 |         |                              |            | 51 |
| 52 |         |                              |            | 52 |
| 53 |         |                              |            | 53 |
| 54 |         |                              |            | 54 |
| 55 |         |                              |            | 55 |
| 56 |         |                              |            | 56 |
| 57 |         |                              |            | 57 |
| 58 |         |                              |            | 58 |
| 59 |         |                              |            | 59 |
| 60 |         |                              |            | 60 |

# CONIC\_SUBROUTINES

|    |  |     |              |    |
|----|--|-----|--------------|----|
| 1  |  |     |              | 1  |
| 2  |  |     |              | 2  |
| 3  |  |     |              | 3  |
| 4  |  | BOV | 2PISC        | 4  |
| 5  |  |     | # 2PISC (+6) | 5  |
| 6  |  |     |              | 6  |
| 7  |  |     |              | 7  |
| 8  |  |     |              | 8  |
| 9  |  |     |              | 9  |
| 10 |  |     |              | 10 |
| 11 |  |     |              | 11 |
| 12 |  |     |              | 12 |
| 13 |  |     |              | 13 |
| 14 |  |     |              | 14 |
| 15 |  |     |              | 15 |
| 16 |  |     |              | 16 |
| 17 |  |     |              | 17 |
| 18 |  |     |              | 18 |
| 19 |  |     |              | 19 |
| 20 |  |     |              | 20 |
| 21 |  |     |              | 21 |
| 22 |  |     |              | 22 |
| 23 |  |     |              | 23 |
| 24 |  |     |              | 24 |
| 25 |  |     |              | 25 |
| 26 |  |     |              | 26 |
| 27 |  |     |              | 27 |
| 28 |  |     |              | 28 |
| 29 |  |     |              | 29 |
| 30 |  |     |              | 30 |
| 31 |  |     |              | 31 |
| 32 |  |     |              | 32 |
| 33 |  |     |              | 33 |
| 34 |  |     |              | 34 |
| 35 |  |     |              | 35 |
| 36 |  |     |              | 36 |
| 37 |  |     |              | 37 |
| 38 |  |     |              | 38 |
| 39 |  |     |              | 39 |
| 40 |  |     |              | 40 |
| 41 |  |     |              | 41 |
| 42 |  |     |              | 42 |
| 43 |  |     |              | 43 |
| 44 |  |     |              | 44 |
| 45 |  |     |              | 45 |
| 46 |  |     |              | 46 |
| 47 |  |     |              | 47 |
| 48 |  |     |              | 48 |
| 49 |  |     |              | 49 |
| 50 |  |     |              | 50 |
| 51 |  |     |              | 51 |
| 52 |  |     |              | 52 |
| 53 |  |     |              | 53 |
| 54 |  |     |              | 54 |
| 55 |  |     |              | 55 |
| 56 |  |     |              | 56 |
| 57 |  |     |              | 57 |
| 58 |  |     |              | 58 |
| 59 |  |     |              | 59 |
| 60 |  |     |              | 60 |







|    |                            |       |          |    |
|----|----------------------------|-------|----------|----|
| 1  |                            |       |          | 1  |
| 2  |                            |       |          | 2  |
| 3  | OD                         |       |          | 3  |
| 4  | NEWDELX                    |       |          | 4  |
| 5  | NDXCHNGE                   | DLOAD | DSU      | 5  |
| 6  |                            |       | XMIN     | 6  |
| 7  |                            |       |          | 7  |
| 8  |                            | DMPR  | X        | 8  |
| 9  |                            |       |          | 9  |
| 10 |                            |       | GOTO     | 10 |
| 11 | # TO FORCE MPAC +2 TO ZERO |       |          | 11 |
| 12 |                            |       |          | 12 |
| 13 | DP9/10                     |       |          | 13 |
| 14 | NEWDELX                    |       |          | 14 |
| 15 | POSDELX                    | DLOAD |          | 15 |
| 16 |                            |       |          | 16 |
| 17 | X                          |       |          | 17 |
| 18 |                            | STORE | XMIN     | 18 |
| 19 |                            | BDSU  | DSU      | 19 |
| 20 | # MOVE MIN BOUND IN        |       |          | 20 |
| 21 | #                          |       |          | 21 |
| 22 | PL AT 0                    |       |          | 22 |
| 23 |                            |       |          | 23 |
| 24 |                            | BOV   | XMAX     | 24 |
| 25 |                            |       |          | 25 |
| 26 |                            |       | BMN      | 26 |
| 27 |                            |       |          | 27 |
| 28 |                            |       | PDXCHNGE | 28 |
| 29 |                            |       |          | 29 |
| 30 |                            | DLOAD | PDXCHNGE | 30 |
| 31 |                            |       |          | 31 |
| 32 | OD                         |       |          | 32 |
| 33 | NEWDELX                    | STORE | DELX     | 33 |
| 34 |                            | BZE   | DAD      | 34 |
| 35 |                            |       |          | 35 |
| 36 |                            |       | KEPCONVG | 36 |
| 37 |                            |       |          | 37 |
| 38 |                            | STODL | X        | 38 |
| 39 |                            |       |          | 39 |
| 40 |                            |       | X        | 40 |
| 41 |                            |       |          | 41 |
| 42 |                            |       | T        | 42 |
| 43 | BRNCHCTR                   | STORE | TC       | 43 |
| 44 |                            | RTB   | BHIZ     | 44 |
| 45 |                            |       |          | 45 |
| 46 |                            |       | CHECKCTR | 46 |
| 47 |                            |       |          | 47 |
| 48 |                            |       | KEPCONVG | 48 |
| 49 |                            |       |          | 49 |
| 50 |                            | GOTO  |          | 50 |
| 51 |                            |       |          | 51 |
| 52 |                            |       | KEPLOOP  | 52 |
| 53 | # ITERATE                  |       |          | 53 |
| 54 |                            |       |          | 54 |
| 55 | PDXCHNGE                   | DLOAD | DSU      | 55 |
| 56 |                            |       | XMAX     | 56 |
| 57 |                            |       |          | 57 |
| 58 |                            |       | X        | 58 |
| 59 |                            |       |          | 59 |
| 60 |                            | DMPR  | GOTO     | 60 |
| 61 | # TO FORCE MPAC +2 TO ZERO |       |          | 61 |
| 62 |                            |       |          | 62 |
| 63 | DP9/10                     |       |          | 63 |
| 64 | NEWDELX                    |       |          | 64 |
| 65 | BADX                       | DLOAD | SR1      | 65 |
| 66 |                            |       |          | 66 |
| 67 |                            |       | XMAX     | 67 |
| 68 |                            |       |          | 68 |
| 69 |                            | SIGN  |          | 69 |
| 70 |                            |       |          | 70 |
| 71 |                            |       | TAU.     | 71 |
| 72 |                            |       |          | 72 |
| 73 |                            | STORE | X        | 73 |
| 74 |                            |       |          | 74 |
| 75 |                            | GOTO  |          | 75 |
| 76 |                            |       |          | 76 |
| 77 |                            |       |          | 77 |
| 78 |                            |       |          | 78 |
| 79 |                            |       |          | 79 |
| 80 |                            |       |          | 80 |

|    |                      |       |      |                                       |        |                    |    |
|----|----------------------|-------|------|---------------------------------------|--------|--------------------|----|
| 1  | # CSN10_C00DR00TINES |       |      | PAGE 1170                             |        |                    | 1  |
| 2  |                      |       |      | STORBND                               |        |                    | 2  |
| 3  | TIMEOVFL             | DLOAD | BMN  | # X WAS TOO BIG                       |        |                    | 3  |
| 4  |                      |       |      | X                                     |        |                    | 5  |
| 5  |                      |       |      | NEGTOVFL                              |        |                    | 6  |
| 6  |                      |       |      | XMAX                                  |        |                    | 7  |
| 7  | CMNTOVFL             | DLOAD | SR1  |                                       |        |                    | 9  |
| 8  |                      |       |      | DELX                                  |        |                    | 10 |
| 9  |                      |       |      | DELX                                  |        |                    | 11 |
| 10 |                      |       |      | BZ                                    |        |                    | 13 |
| 11 |                      |       |      | KEPRTN                                |        |                    | 14 |
| 12 |                      |       |      | X                                     |        |                    | 15 |
| 13 |                      |       |      | X                                     |        |                    | 17 |
| 14 |                      |       |      | TC                                    |        |                    | 18 |
| 15 |                      |       |      | T                                     |        |                    | 19 |
| 16 |                      |       |      | GOTO                                  |        |                    | 21 |
| 17 |                      |       |      | BRNCHCTR                              |        |                    | 22 |
| 18 | NEGTOVFL             | STORE | XMIN |                                       |        |                    | 23 |
| 19 |                      |       |      | GOTO                                  |        |                    | 24 |
| 20 |                      |       |      | CMNTOVFL                              |        |                    | 25 |
| 21 | KEPCONVG             | DLOAD | SR4R |                                       |        |                    | 26 |
| 22 |                      |       |      | R1                                    |        |                    | 27 |
| 23 |                      |       |      | VXSC                                  |        |                    | 28 |
| 24 |                      |       |      | XSQC(XI)                              |        |                    | 29 |
| 25 |                      |       |      | URRECT                                |        |                    | 30 |
| 26 | VSL1                 |       | PDDL | # OD=(R1-XSQC(XI))URRECT (+33 OR +31) |        |                    | 31 |
| 27 |                      |       |      | X                                     |        |                    | 32 |
| 28 |                      |       |      | NORM                                  |        |                    | 33 |
| 29 |                      |       |      | X1                                    |        |                    | 34 |
| 30 |                      |       |      | DMPR                                  |        |                    | 35 |
| 31 |                      |       |      | 1/ROOTMU                              |        |                    | 36 |
| 32 |                      |       |      | X                                     |        |                    | 37 |
| 33 |                      |       |      | SRR*                                  |        |                    | 38 |
| 34 |                      |       |      | S(XI)                                 |        |                    | 39 |
| 35 |                      |       |      | 0 -7,1                                |        |                    | 40 |
| 36 |                      |       |      | BDSU                                  |        |                    | 41 |
| 37 |                      |       |      | T                                     |        |                    | 42 |
| 38 |                      |       |      | VXSC                                  |        |                    | 43 |
| 39 |                      |       |      | VRECT                                 |        |                    | 44 |
| 40 |                      |       |      | VAD                                   | #      | PL AT 0            | 45 |
| 41 |                      |       |      | VSL4                                  |        |                    | 46 |
| 42 |                      |       |      | STORE                                 | RCV    | # RCV (+29 OR +27) | 47 |
| 43 |                      |       |      |                                       |        |                    | 48 |
| 44 |                      |       |      | ABVAL                                 | NORM   |                    | 49 |
| 45 |                      |       |      | X2                                    |        |                    | 50 |
| 46 |                      |       |      | STODL                                 | RCNORM |                    | 51 |
| 47 |                      |       |      | XI                                    |        |                    | 52 |
| 48 |                      |       |      | DMPR                                  | DSU    |                    | 53 |
| 49 |                      |       |      | S(XI)                                 |        |                    | 54 |
| 50 |                      |       |      | D1/128                                |        |                    | 55 |
| 51 |                      |       |      |                                       |        |                    | 56 |
| 52 |                      |       |      |                                       |        |                    | 57 |
| 53 |                      |       |      |                                       |        |                    | 58 |
| 54 |                      |       |      |                                       |        |                    | 59 |
| 55 |                      |       |      |                                       |        |                    | 60 |
| 56 |                      |       |      |                                       |        |                    | 61 |
| 57 |                      |       |      |                                       |        |                    | 62 |
| 58 |                      |       |      |                                       |        |                    | 63 |
| 59 |                      |       |      |                                       |        |                    | 64 |
| 60 |                      |       |      |                                       |        |                    | 65 |

|    |   |
|----|---|
| 1  |   |
| 2  |   |
| 3  |   |
| 4  |   |
| 5  |   |
| 6  |   |
| 7  |   |
| 8  |   |
| 9  |   |
| 10 |   |
| 11 |   |
| 12 |   |
| 13 |   |
| 14 |   |
| 15 |   |
| 16 |   |
| 17 |   |
| 18 |   |
| 19 |   |
| 20 |   |
| 21 |   |
| 22 |   |
| 23 |   |
| 24 |   |
| 25 |   |
| 26 |   |
| 27 |   |
| 28 |   |
| 29 |   |
| 30 |   |
| 31 |   |
| 32 |   |
| 33 |   |
| 34 |   |
| 35 |   |
| 36 |   |
| 37 |   |
| 38 |   |
| 39 |   |
| 40 |   |
| 41 |   |
| 42 |   |
| 43 |   |
| 44 |   |
| 45 |   |
| 46 |   |
| 47 |   |
| 48 |   |
| 49 |   |
| 50 |   |
| 51 |   |
| 52 |   |
| 53 |   |
| 54 |   |
| 55 |   |
| 56 |   |
| 57 |   |
| 58 |   |
| 59 |   |
| 60 |   |
| 61 |   |
| 62 |   |
| 63 |   |
| 64 |   |
| 65 |   |
| 66 |   |
| 67 |   |
| 68 |   |
| 69 |   |
| 70 |   |
| 71 |   |
| 72 |   |
| 73 |   |
| 74 |   |
| 75 |   |
| 76 |   |
| 77 | 1 |
| 78 |   |
| 79 |   |



# CONIC\_SUBROUTINES

# MPAC=XI (+6), OD=XSQ (+34 OR +32 -N1)

DELTIME

EXIT

TC

POLY

DEC

8

2DEC

.083333334

2DEC

-.266666684

2DEC

.406349155

2DEC

-.361198675

2DEC

.210153242

2DEC

-.086221951

2DEC

.026268812

2DEC

-.006163316

2DEC

.001177342

2DEC

-.000199055

TC

INTPRET

STODL

S(XI)

XI

EXIT

TC

POLY

DEC

8

2DEC

.031250001

2DEC

-.166666719

2DEC

.355555413

2DEC

-.406347410

2DEC

.288962094

2DEC

-.140117894

2DEC

.049247387

2DEC

-.013081923

2DEC

.002806389

2DEC

-.000529414

TC

INTPRET

|    |                     |       |               |                                             |           |    |
|----|---------------------|-------|---------------|---------------------------------------------|-----------|----|
| 1  | # CSN10 SUBROUTINES |       |               |                                             | PAGE 1101 | 1  |
| 2  |                     | DMP   | SRR*<br>OD    | #                                           | PL AT 0   | 2  |
| 3  |                     |       |               |                                             |           | 3  |
| 4  |                     |       | 0 -5,1        |                                             |           | 4  |
| 5  |                     | STORE | XSQC(XI)      | # XSQC(XI) (+33 OR +31)                     |           | 5  |
| 6  |                     | DMP   | SL1           |                                             |           | 6  |
| 7  |                     |       | KEPC1         |                                             |           | 7  |
| 8  |                     | RTB   | PDDL          | # XCH WITH PL. OD=C1 XSQ C(XI) (+49 OR +46) |           | 8  |
| 9  |                     |       | TPMODE        | #                                           | PL AT 0,3 | 9  |
| 10 |                     | DMP   | SRR*<br>S(XI) |                                             |           | 10 |
| 11 |                     |       | 0 -5,1        |                                             |           | 11 |
| 12 |                     |       |               |                                             |           | 12 |
| 13 |                     | DMP   | SL1           |                                             |           | 13 |
| 14 |                     |       | KEPC2         |                                             |           | 14 |
| 15 |                     | RTB   | PDDL          | # 3D=C2 XSQ S(XI) (+35 OR +33)              | PL AT 6   | 15 |
| 16 |                     |       | TPMODE        |                                             |           | 16 |
| 17 |                     |       | R1            |                                             |           | 17 |
| 18 |                     | SR    | TAD           | #                                           | PL AT 3   | 18 |
| 19 |                     |       | 6             |                                             |           | 19 |
| 20 |                     | NORM  | DMP           | # TO PRESERVE SIGNIF.                       |           | 20 |
| 21 |                     |       | X1            |                                             |           | 21 |
| 22 |                     |       | X             |                                             |           | 22 |
| 23 |                     | SR*   | TAD           | # X(C2 XSQ S(XI) +R1) (+49 OR +46)          | PL AT 0   | 23 |
| 24 |                     |       | 0 -3,1        |                                             |           | 24 |
| 25 |                     | SL4R  | DMPR          |                                             |           | 25 |
| 26 |                     |       | 1/ROOTMU      |                                             |           | 26 |
| 27 |                     | STORE | T             |                                             |           | 27 |
| 28 |                     | RVQ   |               |                                             |           | 28 |
| 29 |                     |       |               |                                             |           | 29 |
| 30 |                     |       |               |                                             |           | 30 |
| 31 |                     |       |               |                                             |           | 31 |
| 32 |                     |       |               |                                             |           | 32 |
| 33 |                     |       |               |                                             |           | 33 |
| 34 |                     |       |               |                                             |           | 34 |
| 35 |                     |       |               |                                             |           | 35 |
| 36 |                     |       |               |                                             |           | 36 |
| 37 |                     |       |               |                                             |           | 37 |
| 38 |                     |       |               |                                             |           | 38 |
| 39 |                     |       |               |                                             |           | 39 |
| 40 |                     |       |               |                                             |           | 40 |
| 41 |                     |       |               |                                             |           | 41 |
| 42 |                     |       |               |                                             |           | 42 |
| 43 |                     |       |               |                                             |           | 43 |
| 44 |                     |       |               |                                             |           | 44 |
| 45 |                     |       |               |                                             |           | 45 |
| 46 |                     |       |               |                                             |           | 46 |
| 47 |                     |       |               |                                             |           | 47 |
| 48 |                     |       |               |                                             |           | 48 |
| 49 |                     |       |               |                                             |           | 49 |
| 50 |                     |       |               |                                             |           | 50 |
| 51 |                     |       |               |                                             |           | 51 |
| 52 |                     |       |               |                                             |           | 52 |
| 53 |                     |       |               |                                             |           | 53 |
| 54 |                     |       |               |                                             |           | 54 |
| 55 |                     |       |               |                                             |           | 55 |
| 56 |                     |       |               |                                             |           | 56 |
| 57 |                     |       |               |                                             |           | 57 |
| 58 |                     |       |               |                                             |           | 58 |
| 59 |                     |       |               |                                             |           | 59 |
| 60 |                     |       |               |                                             |           | 60 |

|    |               |        |           |                                            |           |    |
|----|---------------|--------|-----------|--------------------------------------------|-----------|----|
| 1  | # ORDER CODES |        | PAGE 1102 |                                            | 1         |    |
| 2  | ITERATOR      | BONCLR | DLOAD     |                                            | 2         |    |
| 3  |               |        | SLOPESW   |                                            | 3         |    |
| 4  |               |        | FIRSTIME  |                                            | 4         |    |
| 5  |               | DSU    | DEP       |                                            | 5         |    |
| 6  |               |        | NORM      |                                            | 6         |    |
| 7  |               |        | DEPREV    |                                            | 7         |    |
| 8  |               | PDDL   | X1        |                                            | 8         |    |
| 9  |               |        | NORM      |                                            | 9         |    |
| 10 |               |        | DELINDEP  |                                            | 10        |    |
| 11 |               | XSU,1  | X2        |                                            | 11        |    |
| 12 |               |        | DMP       |                                            | 12        |    |
| 13 |               |        | X2        |                                            | 13        |    |
| 14 |               | SLR*   | DELDEP    |                                            | 14        |    |
| 15 |               |        | DDV       | #                                          | PL UP 2   | 15 |
| 16 |               | SR1    | 1,1       |                                            | 16        |    |
| 17 |               |        | BOFF      |                                            |           | 17 |
| 18 |               |        | ORDERSW   |                                            |           | 18 |
| 19 |               | ABS    | SGNCHECK  |                                            | 19        |    |
| 20 |               |        | SIGN      | # IN CASE 2ND DERIV. CHANGED SIGN, MUST    |           | 20 |
| 21 |               |        | DELDEP    | # DISREGARD IT TO FIND MIN.                |           | 21 |
| 22 | SGNCHECK      | PUSH   | BPL       | # TRIAL DELINDEP                           | 22        |    |
| 23 |               |        |           | POSDEL                                     | PL DOWN 2 | 23 |
| 24 |               | DLOAD  | BON       |                                            | 24        |    |
| 25 |               |        | INDEP     |                                            | 25        |    |
| 26 |               |        | ORDERSW   |                                            | 26        |    |
| 27 |               | STORE  | MINCHECK  |                                            | 27        |    |
| 28 |               |        | MAX       | # IF NOT 2ND ORDER, CAN MOVE MAX BOUND IN. |           | 28 |
| 29 | MINCHECK      | BDSU   | DSU       |                                            | 29        |    |
| 30 |               |        | MIN       |                                            | 30        |    |
| 31 |               | BOV    | BPL       |                                            | 31        |    |
| 32 |               | GOTO   | MODNGDEL  |                                            | 32        |    |
| 33 |               |        | MODNGDEL  |                                            | 33        |    |
| 34 |               |        | DELOK     |                                            | 34        |    |
| 35 | MODNGDEL      | DLOAD  | DSU       | # TRIAL DELINDEP WOULD EXCEED MIN BOUND    | 35        |    |
| 36 |               |        |           | MIN                                        |           | 36 |
| 37 |               | DMP    | INDEP     |                                            | 37        |    |
| 38 |               |        | GOTO      |                                            | 38        |    |
| 39 |               |        | DP9/10    |                                            | 39        |    |
| 40 |               |        | NEWDEL    |                                            | 40        |    |
| 41 | FIRSTIME      | DLOAD  | DMP       |                                            | 41        |    |
| 42 |               |        |           | MIN                                        |           | 42 |
| 43 |               |        | TWEEKIT   | # DLOAD TWEEKIT(40D) SENSITIVE TO CHANGE.  | 43        |    |
| 44 |               | PDDL   | DMP       | # S2(41D) SHOULDNT CONTAIN HI ORDER ONES   | 44        |    |
| 45 |               |        |           |                                            | 45        |    |
| 46 |               |        |           |                                            | 46        |    |
| 47 |               |        |           |                                            | 47        |    |
| 48 |               |        |           |                                            | 48        |    |
| 49 |               |        |           |                                            | 49        |    |
| 50 |               |        |           |                                            | 50        |    |
| 51 |               |        |           |                                            | 51        |    |
| 52 |               |        |           |                                            | 52        |    |
| 53 |               |        |           |                                            | 53        |    |
| 54 |               |        |           |                                            | 54        |    |
| 55 |               |        |           |                                            | 55        |    |
| 56 |               |        |           |                                            | 56        |    |
| 57 |               |        |           |                                            | 57        |    |
| 58 |               |        |           |                                            | 58        |    |
| 59 |               |        |           |                                            | 59        |    |
| 60 |               |        |           |                                            | 60        |    |

|    |          |       |                                            |    |
|----|----------|-------|--------------------------------------------|----|
| 1  |          |       |                                            | 1  |
| 2  |          |       |                                            | 2  |
| 3  |          |       |                                            | 3  |
| 4  |          | DSU   | MAX                                        | 4  |
| 5  |          | SIGN  | TWEEKIT                                    | 5  |
| 6  |          |       |                                            | 6  |
| 7  |          |       | GOTO                                       | 7  |
| 8  |          |       | DELDEP                                     | 8  |
| 9  | POSDEL   | DLOAD | SGNCHECK                                   | 9  |
| 10 |          |       |                                            | 10 |
| 11 |          |       | BON                                        | 11 |
| 12 |          |       | INDEP                                      | 12 |
| 13 |          | STORE | ORDERSW                                    | 13 |
| 14 |          |       | MAXCHECK                                   | 14 |
| 15 | MAXCHECK | BDSU  | MIN                                        | 15 |
| 16 |          |       | # IF NOT 2ND ORDER, CAN MOVE MIN BOUND IN. |    |
| 17 |          | BOV   | DSU                                        | 17 |
| 18 |          |       | MAX                                        | 18 |
| 19 |          |       | BMN                                        | 19 |
| 20 | DELOK    | DLOAD | MODPSDEL                                   | 20 |
| 21 |          |       | MODPSDEL                                   | 21 |
| 22 | NEWDEL   | STORE | OD                                         | 22 |
| 23 |          | RVQ   | DELINDEP                                   | 23 |
| 24 |          |       |                                            | 24 |
| 25 | MODPSDEL | DLOAD | DSU                                        | 25 |
| 26 |          |       | MAX                                        | 26 |
| 27 |          |       | INDEP                                      | 27 |
| 28 |          | DMP   | GOTO                                       | 28 |
| 29 |          |       | DP9/10                                     | 29 |
| 30 |          |       | NEWDEL                                     | 30 |
| 31 | CHECKCTR | CS    | ONE                                        | 31 |
| 32 |          | INDEX | FIXLOC                                     | 32 |
| 33 |          | AD    | ITERCTR                                    | 33 |
| 34 |          | INDEX | FIXLOC                                     | 34 |
| 35 |          | TS    | ITERCTR                                    | 35 |
| 36 |          | TS    | MPAC                                       | 36 |
| 37 |          | TC    | DANZIG                                     | 37 |
| 38 |          |       |                                            | 38 |
| 39 |          |       |                                            | 39 |
| 40 |          |       |                                            | 40 |
| 41 |          |       |                                            | 41 |
| 42 |          |       |                                            | 42 |
| 43 |          |       |                                            | 43 |
| 44 |          |       |                                            | 44 |
| 45 |          |       |                                            | 45 |
| 46 |          |       |                                            | 46 |
| 47 |          |       |                                            | 47 |
| 48 |          |       |                                            | 48 |
| 49 |          |       |                                            | 49 |
| 50 |          |       |                                            | 50 |
| 51 |          |       |                                            | 51 |
| 52 |          |       |                                            | 52 |
| 53 |          |       |                                            | 53 |
| 54 |          |       |                                            | 54 |
| 55 |          |       |                                            | 55 |
| 56 |          |       |                                            | 56 |
| 57 |          |       |                                            | 57 |
| 58 |          |       |                                            | 58 |
| 59 |          |       |                                            | 59 |
| 60 |          |       |                                            | 60 |



|    |          |       |                                           |    |
|----|----------|-------|-------------------------------------------|----|
| 1  |          |       |                                           | 1  |
| 2  | NEWSTATE | DLOAD | SR4R                                      | 2  |
| 3  |          |       | R1                                        | 3  |
| 4  |          | DSU   | VXSC                                      | 4  |
| 5  |          |       | XSQC(XI)                                  | 5  |
| 6  |          |       | UR1                                       | 6  |
| 7  |          | VSL1  | PDDL                                      | 7  |
| 8  |          |       | # OD=(R1-XSQC(XI))UR1 (+33 OR 31) PL AT 6 |    |
| 9  |          | DSQ   | X                                         | 9  |
| 10 |          |       | NORM                                      | 10 |
| 11 |          | DMPR  | X1                                        | 11 |
| 12 |          |       | DMPR                                      | 12 |
| 13 |          |       | 1/ROOTMU                                  | 13 |
| 14 |          | DMP   | X                                         | 14 |
| 15 |          |       | SRR*                                      | 15 |
| 16 |          |       | S(XI)                                     | 16 |
| 17 |          |       | 0 -7,1                                    | 17 |
| 18 |          | BDSU  | T                                         | 18 |
| 19 |          | SL1   | VXSC                                      | 19 |
| 20 |          |       | VVEC                                      | 20 |
| 21 |          | VSL1  | VAD                                       | 21 |
| 22 |          |       | # PL AT 0                                 |    |
| 23 |          | VSL4  | PUSH                                      | 23 |
| 24 | LAMENTER | ABVAL |                                           | 24 |
| 25 |          | NORM  |                                           | 25 |
| 26 |          | STODL | X1                                        | 26 |
| 27 |          |       | R2                                        | 27 |
| 28 |          |       | XI                                        | 28 |
| 29 |          | DMP   | DSU                                       | 29 |
| 30 |          |       | S(XI)                                     | 30 |
| 31 |          |       | D1/128                                    | 31 |
| 32 |          | DMP   | SL1R                                      | 32 |
| 33 |          |       | ROOTMU                                    | 33 |
| 34 |          | DMP   | SLR*                                      | 34 |
| 35 |          |       | X                                         | 35 |
| 36 |          |       | 0 -3,1                                    | 36 |
| 37 |          | DDV   | VXSC                                      | 37 |
| 38 |          |       | R2                                        | 38 |
| 39 |          | VSL1  | UR1                                       | 39 |
| 40 |          |       | PDDL                                      | 40 |
| 41 |          |       | # 6D=V2VEC PART (+15 OR 13) PL AT 12      |    |
| 42 |          |       | XSQC(XI)                                  | 42 |
| 43 |          | SLR*  | DDV                                       | 43 |
| 44 |          |       | 0 -4,1                                    | 44 |
| 45 |          |       | R2                                        | 45 |
| 46 |          | BDSU  |                                           | 46 |
| 47 |          |       | D1/256                                    | 47 |
| 48 |          | VXSC  | VAD                                       | 48 |
| 49 |          |       | # PL AT 6                                 |    |
| 50 |          |       | VVEC                                      | 50 |
| 51 |          | VSL8  | RVQ                                       | 51 |
| 52 |          |       |                                           | 52 |
| 53 |          |       |                                           | 53 |
| 54 |          |       |                                           | 54 |
| 55 |          |       |                                           | 55 |
| 56 |          |       |                                           | 56 |
| 57 |          |       |                                           | 57 |
| 58 |          |       |                                           | 58 |
| 59 |          |       |                                           | 59 |
| 60 |          |       |                                           | 60 |

|    |                                                                   |        |                |                                            |    |
|----|-------------------------------------------------------------------|--------|----------------|--------------------------------------------|----|
| 1  |                                                                   |        |                |                                            | 1  |
| 2  | SETLOC CONICS1                                                    |        |                |                                            | 2  |
| 3  | BANK                                                              |        |                |                                            | 3  |
| 4  |                                                                   |        |                |                                            | 4  |
| 5  | COUNT* \$\$/CONIC                                                 |        |                |                                            | 5  |
| 6  | # DO NOT DISTURB THE ORDER OF THESE CDS, OVERLAYS HAVE BEEN MADE. |        |                |                                            | 6  |
| 7  | BEE17                                                             | DEC    | 0              | # KEEP WITH D1/8 2DEC 1.0B-17 (0000004000) | 7  |
| 8  | D1/8                                                              | 2DEC   | 1.0 B-3        |                                            | 8  |
| 9  | D1/128                                                            | 2DEC   | 1.0 B-7        |                                            | 9  |
| 10 | D1/64                                                             | 2DEC   | 1.0 B-6        |                                            | 10 |
| 11 | D1/4                                                              | 2DEC   | 1.0 B-2        |                                            | 11 |
| 12 | D1/16                                                             | 2DEC   | 1.0 B-4        |                                            | 12 |
| 13 | D1/32                                                             | 2DEC   | 1.0 B-5        |                                            | 13 |
| 14 | D1/1024                                                           | 2DEC   | 1.0 B-10       |                                            | 14 |
| 15 | D1/256                                                            | 2DEC   | 1.0 B-8        |                                            | 15 |
| 16 | DP9/10                                                            | 2DEC   | .9             |                                            | 16 |
| 17 | KEPZERO                                                           | EQUALS | L06ZEROS       |                                            | 17 |
| 18 | -50SC                                                             | 2DEC   | -50.0 B-12     |                                            | 18 |
| 19 | 2PISC                                                             | 2DEC   | 6.28318530 B-6 |                                            | 19 |
| 20 | BEE19                                                             | EQUALS | D1/32 -1       | # 2DEC 1.0 B-19 (00000 01000)              | 20 |
| 21 | BEE22                                                             | EQUALS | D1/256 -1      | # 2DEC 1.0 B-22 (00000 00100)              | 21 |
| 22 | ONEBIT                                                            | 2DEC   | 1.0 B-28       |                                            | 22 |
| 23 | COGUPLIM                                                          | 2DEC   | .999511597     |                                            | 23 |
| 24 | COGLOLIM                                                          | 2DEC   | -.999511597    |                                            | 24 |
| 25 |                                                                   |        |                |                                            | 25 |
| 26 |                                                                   |        |                |                                            | 26 |
| 27 |                                                                   |        |                |                                            | 27 |
| 28 |                                                                   |        |                |                                            | 28 |
| 29 |                                                                   |        |                |                                            | 29 |
| 30 |                                                                   |        |                |                                            | 30 |
| 31 |                                                                   |        |                |                                            | 31 |
| 32 |                                                                   |        |                |                                            | 32 |
| 33 |                                                                   |        |                |                                            | 33 |
| 34 |                                                                   |        |                |                                            | 34 |
| 35 |                                                                   |        |                |                                            | 35 |
| 36 |                                                                   |        |                |                                            | 36 |
| 37 |                                                                   |        |                |                                            | 37 |
| 38 |                                                                   |        |                |                                            | 38 |
| 39 |                                                                   |        |                |                                            | 39 |
| 40 |                                                                   |        |                |                                            | 40 |
| 41 |                                                                   |        |                |                                            | 41 |
| 42 |                                                                   |        |                |                                            | 42 |
| 43 |                                                                   |        |                |                                            | 43 |
| 44 |                                                                   |        |                |                                            | 44 |
| 45 |                                                                   |        |                |                                            | 45 |
| 46 |                                                                   |        |                |                                            | 46 |
| 47 |                                                                   |        |                |                                            | 47 |
| 48 |                                                                   |        |                |                                            | 48 |
| 49 |                                                                   |        |                |                                            | 49 |
| 50 |                                                                   |        |                |                                            | 50 |
| 51 |                                                                   |        |                |                                            | 51 |
| 52 |                                                                   |        |                |                                            | 52 |
| 53 |                                                                   |        |                |                                            | 53 |
| 54 |                                                                   |        |                |                                            | 54 |
| 55 |                                                                   |        |                |                                            | 55 |
| 56 |                                                                   |        |                |                                            | 56 |
| 57 |                                                                   |        |                |                                            | 57 |
| 58 |                                                                   |        |                |                                            | 58 |
| 59 |                                                                   |        |                |                                            | 59 |
| 60 |                                                                   |        |                |                                            | 60 |

|    |          |        |                                 |    |   |
|----|----------|--------|---------------------------------|----|---|
| 1  | CONICS   |        |                                 |    | 1 |
| 2  |          | SETLOC | CONICS                          | 2  |   |
| 3  |          | BANK   |                                 | 3  |   |
| 4  |          |        |                                 | 4  |   |
| 5  |          | COUNT* | \$\$/CONIC                      | 5  |   |
| 6  | TIMETHET | STQ    | SETPD                           | 6  |   |
| 7  |          |        | #                               | 7  |   |
| 8  |          |        | PL AT 0                         | 8  |   |
| 9  |          |        | RTNTT                           | 9  |   |
| 10 |          | BOV    | 0                               | 10 |   |
| 11 |          |        |                                 | 11 |   |
| 12 |          | VLOAD  | +1                              | 12 |   |
| 13 |          |        | PDL                             | 13 |   |
| 14 |          |        | RVEC                            | 14 |   |
| 15 |          | CALL   | VVEC                            | 15 |   |
| 16 |          |        |                                 | 16 |   |
| 17 |          | BOV    | CALL                            | 17 |   |
| 18 |          |        | COGAOVFL                        | 18 |   |
| 19 | COMMNOUT | DLOAD  | GETX                            | 19 |   |
| 20 |          |        | BON                             | 20 |   |
| 21 |          |        | XI                              | 21 |   |
| 22 |          |        | INFINFLG                        | 22 |   |
| 23 |          | CLEAR  | ABTCONIC                        | 23 |   |
| 24 |          |        | CALL                            | 24 |   |
| 25 |          |        | COGAFLAG                        | 25 |   |
| 26 |          | BON    | DELTIME                         | 26 |   |
| 27 |          |        | CALL                            | 27 |   |
| 28 |          |        | RVS                             | 28 |   |
| 29 |          |        | RTNTT                           | 29 |   |
| 30 |          | GOTO   | NEWSTATE                        | 30 |   |
| 31 |          |        |                                 | 31 |   |
| 32 |          |        | RTNTT                           | 32 |   |
| 33 | COGAOVFL | SETGO  |                                 | 33 |   |
| 34 |          |        |                                 | 34 |   |
| 35 |          |        | COGAFLAG                        | 35 |   |
| 36 |          |        | ABTCONIC                        | 36 |   |
| 37 |          | BANK   | 4                               | 37 |   |
| 38 |          | SETLOC | CONICS1                         | 38 |   |
| 39 |          | BANK   |                                 | 39 |   |
| 40 | PARAM    | COUNT* | \$\$/CONIC                      | 40 |   |
| 41 |          | STQ    | CLEAR                           | 41 |   |
| 42 |          |        | # MPAC=V1VEC, OD=R1VEC          | 42 |   |
| 43 |          |        | PL AT 6                         | 43 |   |
| 44 |          |        | RTNPRM                          | 44 |   |
| 45 |          | CLEAR  | NORMSW                          | 45 |   |
| 46 |          |        |                                 | 46 |   |
| 47 |          | SSP    | COGAFLAG                        | 47 |   |
| 48 |          |        | CALL                            | 48 |   |
| 49 |          |        | GEOMSGN                         | 49 |   |
| 50 |          |        | 37777                           | 50 |   |
| 51 |          |        | # GAMMA ALWAYS LESS THAN 180DEG | 51 |   |
| 52 |          |        | GEOM                            | 52 |   |
| 53 |          |        | # MPAC=SGA (+1), OD=CSGA (+1)   | 53 |   |
| 54 |          | STODL  | 36D                             | 54 |   |
| 55 |          | SR     | DDV                             | 55 |   |
| 56 |          |        | # 36D=SIN GAMMA (+1)            | 56 |   |
| 57 |          |        | PL AT 0                         | 57 |   |
| 58 |          |        |                                 | 58 |   |
| 59 |          |        |                                 | 59 |   |
| 60 |          |        |                                 | 60 |   |

14121HE

# CONIC\_SUBROUTINES

|    |          |       |          |                                      |         |    |
|----|----------|-------|----------|--------------------------------------|---------|----|
| 1  |          |       |          |                                      |         | 1  |
| 2  | GEOM     | UNIT  |          | # MPAC=V2VEC, OD=R1VEC               | PL AT 6 | 2  |
| 3  |          | STODL | U2       | # U2 (+1)                            |         | 3  |
| 4  |          |       | 36D      |                                      |         | 4  |
| 5  |          | STOVL | MAGVEC2  | #                                    | PL AT 0 | 5  |
| 6  |          | UNIT  |          |                                      |         | 6  |
| 7  |          | STORE | UR1      | # UR1 (+1)                           |         | 7  |
| 8  |          | DOT   | SL1      |                                      |         | 8  |
| 9  |          |       | U2       |                                      |         | 9  |
| 10 |          | PDDL  |          | # OD=CSTH (+1)                       | PL AT 2 | 10 |
| 11 |          |       | 36D      |                                      |         | 11 |
| 12 |          | STOVL | R1       | # R1 (+29 OR +27)                    |         | 12 |
| 13 |          |       | UR1      |                                      |         | 13 |
| 14 |          | VXV   | VSL1     |                                      |         | 14 |
| 15 |          |       | U2       |                                      |         | 15 |
| 16 |          | BON   | SIGN     |                                      |         | 16 |
| 17 |          |       | NORMSW   |                                      |         | 17 |
| 18 |          |       | HAVENORM |                                      |         | 18 |
| 19 |          |       | GEOMSGN  |                                      |         | 19 |
| 20 |          | UNIT  | BOV      |                                      |         | 20 |
| 21 |          |       | COLINEAR |                                      |         | 21 |
| 22 | UNITNORM | STODL | UN       | # UN (+1)                            |         | 22 |
| 23 |          |       | 36D      |                                      |         | 23 |
| 24 |          | SIGN  | RVQ      | # MPAC=SNTH (+1), 34D=SNTH.SNTH (+2) |         | 24 |
| 25 |          |       | GEOMSGN  |                                      |         | 25 |
| 26 |          |       |          |                                      |         | 26 |
| 27 | COLINEAR | VSR1  | GOTO     |                                      |         | 27 |
| 28 |          |       | UNITNORM |                                      |         | 28 |
| 29 |          |       |          |                                      |         | 29 |
| 30 | HAVENORM | ABVAL | SIGN     |                                      |         | 30 |
| 31 |          |       | GEOMSGN  |                                      |         | 31 |
| 32 |          | RVQ   |          | # MPAC=SNTH (+1), 34D=SNTH.SNTH (+2) |         | 32 |
| 33 |          |       |          |                                      |         | 33 |
| 34 |          |       |          |                                      |         | 34 |
| 35 |          |       |          |                                      |         | 35 |
| 36 |          |       |          |                                      |         | 36 |
| 37 |          |       |          |                                      |         | 37 |
| 38 |          |       |          |                                      |         | 38 |
| 39 |          |       |          |                                      |         | 39 |
| 40 |          |       |          |                                      |         | 40 |
| 41 |          |       |          |                                      |         | 41 |
| 42 |          |       |          |                                      |         | 42 |
| 43 |          |       |          |                                      |         | 43 |
| 44 |          |       |          |                                      |         | 44 |
| 45 |          |       |          |                                      |         | 45 |
| 46 |          |       |          |                                      |         | 46 |
| 47 |          |       |          |                                      |         | 47 |
| 48 |          |       |          |                                      |         | 48 |
| 49 |          |       |          |                                      |         | 49 |
| 50 |          |       |          |                                      |         | 50 |
| 51 |          |       |          |                                      |         | 51 |
| 52 |          |       |          |                                      |         | 52 |
| 53 |          |       |          |                                      |         | 53 |
| 54 |          |       |          |                                      |         | 54 |
| 55 |          |       |          |                                      |         | 55 |
| 56 |          |       |          |                                      |         | 56 |
| 57 |          |       |          |                                      |         | 57 |
| 58 |          |       |          |                                      |         | 58 |
| 59 |          |       |          |                                      |         | 59 |
| 60 |          |       |          |                                      |         | 60 |

|    |          |        |            |                            |          |
|----|----------|--------|------------|----------------------------|----------|
| 1  |          |        |            |                            |          |
| 2  |          | BANK   | 12         |                            |          |
| 3  |          | SETLOC | CONICS     |                            |          |
| 4  |          | BANK   |            |                            |          |
| 5  |          |        |            |                            |          |
| 6  |          | COUNT* | \$\$/CONIC |                            |          |
| 7  | GETX     | AXT,2  | SSP        | # ASSUMES P (+4) IN MPAC   |          |
| 8  |          |        | 3          |                            |          |
| 9  |          |        | S2         |                            |          |
| 10 |          |        | 1          |                            |          |
| 11 |          | CLEAR  |            |                            |          |
| 12 |          |        | 360SW      |                            |          |
| 13 |          | SQRT   | PDDL       | # OD=SQRT(P)               | PL AT 2  |
| 14 |          |        | CSTH       |                            |          |
| 15 |          | SR1    | BDSU       |                            |          |
| 16 |          |        | D1/4       |                            |          |
| 17 |          | PDDL   | SRR        | #                          | PL AT 4D |
| 18 |          |        | SNTH       |                            |          |
| 19 |          |        | 6          |                            |          |
| 20 |          | DDV    |            | #                          | PL AT 2  |
| 21 |          | BOV    |            |                            |          |
| 22 |          |        | 360CHECK   |                            |          |
| 23 |          | DSU    | DMP        |                            |          |
| 24 |          |        | COGA       | #                          | PL AT 0  |
| 25 |          | SL2R   | BOV        |                            |          |
| 26 |          |        | 360CHECK   |                            |          |
| 27 | WLOOP    | PUSH   | DSQ        | # OD=W (+5)                | PL AT 2  |
| 28 |          | TLOAD  | PDDL       | # 2D=WSQ (+10)             | PL AT 5  |
| 29 |          |        | MPAC       |                            |          |
| 30 |          |        | R1A        |                            |          |
| 31 |          | SR4    | TAD        | #                          | PL AT 2  |
| 32 |          | BMN    | SQRT       |                            |          |
| 33 |          |        | INFINITY   |                            |          |
| 34 |          | ROUND  | DAD        | #                          | PL AT 0D |
| 35 |          | BOV    | TIX,2      |                            |          |
| 36 |          |        | RESETX2    |                            |          |
| 37 |          |        | WLOOP      |                            |          |
| 38 |          |        |            |                            |          |
| 39 |          | BDDV   | BOV        |                            |          |
| 40 |          |        | D1/128     |                            |          |
| 41 |          |        | INFINITY   |                            |          |
| 42 | POLYCOEF | BMN    | PUSH       | # OD=1/W (+2) OR 16/W (+6) | PL AT 2  |
| 43 |          |        | INFINITY   |                            |          |
| 44 |          | DSQ    |            |                            |          |
| 45 |          | NORM   | DMP        |                            |          |
| 46 |          |        | X1         |                            |          |
| 47 |          |        | R1A        |                            |          |
| 48 |          | SRR*   | EXIT       |                            |          |
| 49 |          |        | 0 -10D,1   |                            |          |
| 50 |          | TC     | POLY       |                            |          |
| 51 |          |        |            |                            |          |
| 52 |          |        |            |                            |          |
| 53 |          |        |            |                            |          |
| 54 |          |        |            |                            |          |
| 55 |          |        |            |                            |          |
| 56 |          |        |            |                            |          |
| 57 |          |        |            |                            |          |
| 58 |          |        |            |                            |          |
| 59 |          |        |            |                            |          |
| 60 |          |        |            |                            |          |

1412THE 1

RESETX2

AXT,2

3

360CHECK

SETPD

BPL

OD

INVRSEQN

SET

360SW

INVRSEQN

DLOAD

SQRT

P

PDDL

DMP

# OD=SQRT(P) (+2)

PL AT 2

SNTH

COGA

SL1

PDDL

# 2D=SNTH COGA (+5)

PL AT 4

CSTH

SR4

DAD

D1/32

DSU

DMP

#

PL AT 2,0

NORM

BDDV

X1

SLR\*

SNTH

ABS

# NOTE: NEAR 360 CASE TREATED DIFFERENTLY

PUSH

0 -5,1

STODL

DSQ

# OD=1/W (-1)

PL AT 2

34D

1/WLOOP

PUSH

D1/16

RTB

DSQ

# 2D=G (+4)

PL AT 4

PDDL

#

PL AT 7

DMP

TPMODE

R1A

SR4

34D

TAD

#

PL AT 4

BMN

SQRT

DAD

INFINITY

#

PL AT 2

TIX,2

NORM

1/WLOOP

BDDV

X1

SLR\*

GOTO

#

PL AT 0

0 -7,1

POLYCOEF

TRUE360X

DLOAD

BMN

R1A



[illegible]

LAMBERT

STQ

SETPD  
RTNLAMB  
OD

BOV

+1

CLEAR

VLOAD\*  
SOLNSW  
MUTABLE,1

STODL

1/MU  
TDESIRED

DMPR

STORE  
SETBEE19  
EPSILONL  
VLOAD

PDVL

SLOPESW  
R1VEC  
CALL # OD=R1VEC (+29 OR +27) PL AT 6

STODL

R2VEC # MPAC=R2VEC (+29 OR +27)  
GEOM  
SNTH # OD=CSTH (+1) PL AT 2

NORM

MAGVEC2  
PDDL  
X1

SR1

R1  
DDV

SL\*

PDDL # DXCH WITH OD, OD=R1/R2 (+7) PL AT 0,2  
0 -6,1

STADR

STORE

CSTH # CSTH (+1)

SR1

BDSU  
D1/4

STORE

1-CSTH # 1-CSTH (+2)

ROUND

BZE  
360LAMB

NORM

PDDL # PL AT 4  
X1  
OD

SR1

DDV

SL\*

SQRT # PL AT 2  
0 -3,1

PDDL

SR # 2D=SQRT(2R1/R2(1-CSTH)) (+5) PL AT 4  
SNTH  
6

DDV

DAD # PL AT 2  
1-CSTH

STADR

STORE

COGAMAX

BOV

BMN # IF OVFL, COGAMAX=COGUPLIM  
UPLIM # IF NEG, USE EVEN IF LT COGLOLIM, SINCE

|    |          |       |          |   |                                          |
|----|----------|-------|----------|---|------------------------------------------|
| 1  |          |       |          |   |                                          |
| 2  |          |       | MAXCOGA  | # | THIS WOULD BE RESET IN LAMBLOOP          |
| 3  |          | DSU   | BMN      | # | IF COGAMAX GT COGUPLIM, COGAMAX=COGUPLIM |
| 4  |          |       | COGUPLIM |   |                                          |
| 5  |          |       | MAXCOGA  | # | OTHERWISE OK, SO GO TO MAXCOGA           |
| 6  | UPLIM    | DLOAD |          |   |                                          |
| 7  |          |       | COGUPLIM | # | COGUPLIM=.999511597 = MAX VALUE OF COGA  |
| 8  |          | STORE | COGAMAX  | # | NOT CAUSING OVFL IN R1A CALCULATION      |
| 9  | MAXCOGA  | DLOAD |          |   |                                          |
| 10 |          |       | CSTH     |   |                                          |
| 11 |          | SR    | DSU      | # | PL AT 0                                  |
| 12 |          |       | 6        |   |                                          |
| 13 |          | STADR |          |   |                                          |
| 14 |          | STODL | CSTH-RHO |   |                                          |
| 15 |          |       | GEOMSGN  |   |                                          |
| 16 |          | BMN   | DLOAD    |   |                                          |
| 17 |          |       | LOLIM    |   |                                          |
| 18 |          |       | CSTH-RHO |   |                                          |
| 19 |          | SL1   | DDV      |   |                                          |
| 20 |          |       | SNTH     |   |                                          |
| 21 |          | BOV   |          |   |                                          |
| 22 |          |       | LOLIM    |   |                                          |
| 23 | MINCOGA  | STORE | COGAMIN  | # | COGAMIN (+5)                             |
| 24 |          | BON   | SSP      |   |                                          |
| 25 |          |       | GUESSW   |   |                                          |
| 26 |          |       | NOGUESS  |   |                                          |
| 27 |          |       | TWEEKIT  |   |                                          |
| 28 |          |       | 00001    |   |                                          |
| 29 |          | DLOAD |          |   |                                          |
| 30 |          |       | COGA     |   |                                          |
| 31 |          |       |          |   |                                          |
| 32 | LAMBLOOP | DMP   |          |   |                                          |
| 33 |          |       | SNTH     |   |                                          |
| 34 |          | SR1   | DSU      |   |                                          |
| 35 |          |       | CSTH-RHO |   |                                          |
| 36 |          | NORM  | PDDL     | # | OD=SNTH COGA-(CSTH-RHO) (+7+C(X1)) PL=2  |
| 37 |          |       | X1       |   |                                          |
| 38 |          |       | 1-CSTH   |   |                                          |
| 39 |          | SL*   | DDV      | # | 1-CSTH (+2) PL AT 0                      |
| 40 |          |       | 0 -9D,1  |   |                                          |
| 41 |          | BMN   | BZE      |   |                                          |
| 42 |          |       | NEGP     |   |                                          |
| 43 |          |       | NEGP     |   |                                          |
| 44 |          | STODL | P        | # | P=(1-CSTH)/(SNTH COGA-(CSTH-RHO)) (+4)   |
| 45 |          |       | COGA     |   |                                          |
| 46 |          | DSQ   | DAD      |   |                                          |
| 47 |          |       | D1/1024  |   |                                          |
| 48 |          | NORM  | DMP      |   |                                          |
| 49 |          |       | X1       |   |                                          |
| 50 |          |       | P        |   |                                          |
| 51 |          |       |          |   |                                          |
| 52 |          |       |          |   |                                          |
| 53 |          |       |          |   |                                          |
| 54 |          |       |          |   |                                          |
| 55 |          |       |          |   |                                          |
| 56 |          |       |          |   |                                          |
| 57 |          |       |          |   |                                          |
| 58 |          |       |          |   |                                          |
| 59 |          |       |          |   |                                          |
| 60 |          |       |          |   |                                          |

[illegible]

PL AT 0

#

1412THE

|    |          |       |           |                        |    |
|----|----------|-------|-----------|------------------------|----|
| 1  |          |       |           |                        | 1  |
| 2  | LAMBLOOP |       |           |                        | 2  |
| 3  |          |       |           |                        | 3  |
| 4  | LOLIM    | DLOAD | GOTO      |                        | 4  |
| 5  |          |       | COGLOLIM  | # COGLOLIM=-.999511597 | 5  |
| 6  |          |       | MINCOGA   |                        | 6  |
| 7  |          |       |           |                        | 7  |
| 8  | INITV    | DLOAD | NORM      |                        | 8  |
| 9  |          |       | R1        |                        | 9  |
| 10 |          |       | X1        |                        | 10 |
| 11 |          | PDDL  | SR1       | #                      | 11 |
| 12 |          |       | P         |                        | 12 |
| 13 |          | DDV   |           | #                      | 13 |
| 14 |          | SL*   | SQRT      |                        | 14 |
| 15 |          |       | 0 -4,1    |                        | 15 |
| 16 |          | DMP   | SL1       |                        | 16 |
| 17 |          |       | ROOTMU    |                        | 17 |
| 18 |          | PUSH  | DMP       | # OD=VTAN (+7)         | 18 |
| 19 |          |       | COGA      |                        | 19 |
| 20 |          | SL    | VXSC      |                        | 20 |
| 21 |          |       | 5         |                        | 21 |
| 22 |          |       | UR1       |                        | 22 |
| 23 |          | PDDL  |           | # XCH WITH OD          | 23 |
| 24 |          | VXSC  | VSL1      |                        | 24 |
| 25 |          |       | UN        |                        | 25 |
| 26 |          | VXV   | VAD       | #                      | 26 |
| 27 |          |       | UR1       |                        | 27 |
| 28 |          | VSL1  |           |                        | 28 |
| 29 |          | STORE | VVEC      |                        | 29 |
| 30 |          | SLOAD | BZE       |                        | 30 |
| 31 |          |       | VTARGETAG |                        | 31 |
| 32 |          |       | TARGETV   |                        | 32 |
| 33 |          | GOTO  |           |                        | 33 |
| 34 |          |       | RTNLAMB   |                        | 34 |
| 35 | TARGETV  | DLOAD | CALL      |                        | 35 |
| 36 |          |       | MAGVEC2   |                        | 36 |
| 37 |          |       | LAMENTER  |                        | 37 |
| 38 |          | STORE | VTARGET   |                        | 38 |
| 39 |          | GOTO  |           |                        | 39 |
| 40 |          |       | RTNLAMB   |                        | 40 |
| 41 |          |       |           |                        | 41 |
| 42 |          |       |           |                        | 42 |
| 43 |          |       |           |                        | 43 |
| 44 |          |       |           |                        | 44 |
| 45 |          |       |           |                        | 45 |
| 46 |          |       |           |                        | 46 |
| 47 |          |       |           |                        | 47 |
| 48 |          |       |           |                        | 48 |
| 49 |          |       |           |                        | 49 |
| 50 |          |       |           |                        | 50 |
| 51 |          |       |           |                        | 51 |
| 52 |          |       |           |                        | 52 |
| 53 |          |       |           |                        | 53 |
| 54 |          |       |           |                        | 54 |
| 55 |          |       |           |                        | 55 |
| 56 |          |       |           |                        | 56 |
| 57 |          |       |           |                        | 57 |
| 58 |          |       |           |                        | 58 |
| 59 |          |       |           |                        | 59 |
| 60 |          |       |           |                        | 60 |

# CONIC\_SUBROUTINES

|    |         |       |          |                     |    |
|----|---------|-------|----------|---------------------|----|
| 1  |         |       |          |                     | 1  |
| 2  | TIMERAD | STQ   | SETPD    | #                   | 2  |
| 3  |         |       | RTNTR    |                     | 3  |
| 4  |         |       | 0        |                     | 4  |
| 5  |         | BOV   |          |                     | 5  |
| 6  |         |       | +1       |                     | 6  |
| 7  |         | VLOAD | PDVL     | #                   | 7  |
| 8  |         |       | RVEC     |                     | 8  |
| 9  |         |       | VVEC     |                     | 9  |
| 10 |         | CALL  |          |                     | 10 |
| 11 |         |       | PARAM    |                     | 11 |
| 12 |         | BOV   | DLOAD    | #                   | 12 |
| 13 |         |       | COGAOVFL |                     | 13 |
| 14 |         |       | D1/32    |                     | 14 |
| 15 |         | DSU   | DMP      |                     | 15 |
| 16 |         |       | R1A      |                     | 16 |
| 17 |         |       | P        |                     | 17 |
| 18 |         | SQRT  | DMP      |                     | 18 |
| 19 |         |       | COGA     |                     | 19 |
| 20 |         | SL4   | VXSC     |                     | 20 |
| 21 |         |       | U2       |                     | 21 |
| 22 |         | PDDL  | DSU      | #                   | 22 |
| 23 |         |       | D1/64    |                     | 23 |
| 24 |         |       | R1A      |                     | 24 |
| 25 |         | VXSC  | VSU      | #                   | 25 |
| 26 |         |       | UR1      |                     | 26 |
| 27 |         | VSL4  | UNIT     |                     | 27 |
| 28 |         | BOV   |          |                     | 28 |
| 29 |         |       | CIRCULAR |                     | 29 |
| 30 |         | PDDL  | NORM     | # OD=UNIT(ECC) (+3) | 30 |
| 31 |         |       | RDESIRED | # 35D=ECC (+3)      | 31 |
| 32 |         |       | X1       |                     | 32 |
| 33 |         | PDDL  | DMP      | #                   | 33 |
| 34 |         |       | R1       |                     | 34 |
| 35 |         |       | P        |                     | 35 |
| 36 |         | SL*   | DDV      | #                   | 36 |
| 37 |         |       | 0,1      |                     | 37 |
| 38 |         | DSU   | DDV      |                     | 38 |
| 39 |         |       | D1/16    |                     | 39 |
| 40 |         |       | 36D      | # 36D=ECC (+3)      | 40 |
| 41 |         | STORE | COSF     |                     | 41 |
| 42 |         | BOV   | DSQ      |                     | 42 |
| 43 |         |       | BADR2    |                     | 43 |
| 44 |         | BDSU  | BMN      |                     | 44 |
| 45 |         |       | D1/4     |                     | 45 |
| 46 |         |       | BADR2    |                     | 46 |
| 47 |         | SQRT  | SIGN     |                     | 47 |
| 48 |         |       | SGNRDOT  |                     | 48 |
| 49 |         | CLEAR |          |                     | 49 |
| 50 |         |       | APSESW   |                     | 50 |
| 51 |         |       |          |                     | 51 |
| 52 |         |       |          |                     | 52 |
| 53 |         |       |          |                     | 53 |
| 54 |         |       |          |                     | 54 |
| 55 |         |       |          |                     | 55 |
| 56 |         |       |          |                     | 56 |
| 57 |         |       |          |                     | 57 |
| 58 |         |       |          |                     | 58 |
| 59 |         |       |          |                     | 59 |
| 60 |         |       |          |                     | 60 |

1412THE

1



1412 THE



|    |          |      |        |    |
|----|----------|------|--------|----|
| 1  |          |      |        | 1  |
| 2  | ABTCONIC | EXIT |        | 2  |
| 3  |          | TC   | POOD00 | 3  |
| 4  |          | OCT  | 00607  | 4  |
| 5  |          |      |        | 5  |
| 6  |          |      |        | 6  |
| 7  |          |      |        | 7  |
| 8  |          |      |        | 8  |
| 9  |          |      |        | 9  |
| 10 |          |      |        | 10 |
| 11 |          |      |        | 11 |
| 12 |          |      |        | 12 |
| 13 |          |      |        | 13 |
| 14 |          |      |        | 14 |
| 15 |          |      |        | 15 |
| 16 |          |      |        | 16 |
| 17 |          |      |        | 17 |
| 18 |          |      |        | 18 |
| 19 |          |      |        | 19 |
| 20 |          |      |        | 20 |
| 21 |          |      |        | 21 |
| 22 |          |      |        | 22 |
| 23 |          |      |        | 23 |
| 24 |          |      |        | 24 |
| 25 |          |      |        | 25 |
| 26 |          |      |        | 26 |
| 27 |          |      |        | 27 |
| 28 |          |      |        | 28 |
| 29 |          |      |        | 29 |
| 30 |          |      |        | 30 |
| 31 |          |      |        | 31 |
| 32 |          |      |        | 32 |
| 33 |          |      |        | 33 |
| 34 |          |      |        | 34 |
| 35 |          |      |        | 35 |
| 36 |          |      |        | 36 |
| 37 |          |      |        | 37 |
| 38 |          |      |        | 38 |
| 39 |          |      |        | 39 |
| 40 |          |      |        | 40 |
| 41 |          |      |        | 41 |
| 42 |          |      |        | 42 |
| 43 |          |      |        | 43 |
| 44 |          |      |        | 44 |
| 45 |          |      |        | 45 |
| 46 |          |      |        | 46 |
| 47 |          |      |        | 47 |
| 48 |          |      |        | 48 |
| 49 |          |      |        | 49 |
| 50 |          |      |        | 50 |
| 51 |          |      |        | 51 |
| 52 |          |      |        | 52 |
| 53 |          |      |        | 53 |
| 54 |          |      |        | 54 |
| 55 |          |      |        | 55 |
| 56 |          |      |        | 56 |
| 57 |          |      |        | 57 |
| 58 |          |      |        | 58 |
| 59 |          |      |        | 59 |
| 60 |          |      |        | 60 |

# CONIC\_SUBROUTINES

LDPOSMAX            EQUALS    LODPMAX            # DPPOSMAX IN LOW MEMORY.

# ERASABLE ASSIGNMENTS

# KEPLER SUBROUTINE

#            INPUT -

# RRECT            ERASE    +5

# VRECT            ERASE    +5

# TAU.            ERASE    +1

# XKEP            ERASE    +1

# TC            ERASE    +1

# XPREV            ERASE    +1

1/MU            EQUALS    14D

ROOTMU            EQUALS    16D

1/ROOTMU            EQUALS    18D

#            OUTPUT -

# RCV            ERASE    +5

# VCV            ERASE    +5

# RC            ERASE    +1

# XPREV            ERASE    +1

#            DEBRIS -

ALPHA            EQUALS    8D

XMAX            EQUALS    10D

XMIN            EQUALS    12D

X            EQUALS    20D

XI            EQUALS    24D

S(XI)            EQUALS    26D

XSQC(XI)            EQUALS    28D

T            EQUALS    30D

R1            EQUALS    32D

KEPC1            EQUALS    34D

KEPC2            EQUALS    36D

# DELX            ERASE    +1

# DELT            ERASE    +1

# URRECT            ERASE    +5

# RCNORM            ERASE    +1

# XPREV            EQUALS    XKEP

# LAMBERT SUBROUTINE

#

#            INPUT -

# R1VEC            ERASE    +5

# R2VEC            ERASE    +5

# TDESIRED            ERASE    +1

# GEOMSGN            ERASE    +0

# GUESSW                            # 0 IF COGA GUESS AVIABLE, 1 IF NOT

```
1 # COGA ERASE +1 # INPUT ONLY IF GUESS IS ZERO.
2 # NORMSW # 0 IF UN TO BE COMPUTED, 1 IF UN INPUT
3
4 # UN ERASE +5 # ONLY USED IF NORMSW IS 1
5 # VTARGETAG ERASE +0
6 # TWEKIT EQUALS 40D # ONLY USED IF GUESSW IS 0
7
8 # OUTPUT -
9 # VTARGET ERASE +5 # AVAILABLE ONLY IF VTARGETAG IS ZERO.
10 # VIVEC EQUALS MPAC
11
12 # DEBRIS -
13 # RTNLAMB ERASE +0
14 # U2 ERASE +5
15 # MAGVEC2 ERASE +1
16 # UR1 ERASE +5
17 # R1 EQUALS 31D
18 # UN ERASE +5
19 # SNTH ERASE +1
20 # CSTH ERASE +1
21 # 1-CSTH ERASE +1
22 # CSTH-RHO ERASE +1
23 COGAMAX EQUALS 14D # CLOBBERS 1/MU
24 COGAMIN EQUALS 8D
25 DCOGA EQUALS 12D
26 # TWEKIT EQUALS 40D
27 # P ERASE +1
28 # COGA ERASE +1
29 # R1A ERASE +1
30 # X EQUALS 20D
31 # XSQ EQUALS 22D
32 # XI EQUALS 24D
33 # S(XI) EQUALS 26D
34 # XSQC(XI) EQUALS 28D
35 # T EQUALS 30D
36 # KEPC1 EQUALS 34D
37 # KEPC2 EQUALS 36D
38 # SLOPESW
39 # SOLNSW
40
41 # OTHERS -
42 # RVEC EQUALS R1VEC
43 # VVEC ERASE +5
44 # COGAFLAG
45 # RVSW
46 # INFINFLG
47 # APSESW
48 # 360SW
49 # RTNTT EQUALS RTNLAMB
50 # ECC ERASE +1
51 # RTNTR EQUALS RTNLAMB
```

|            |        |             |                         |
|------------|--------|-------------|-------------------------|
| # EPSILONL | EQUALS | EPSILONT +2 | # DOUBLE PRECISION WORD |
|------------|--------|-------------|-------------------------|

## # 1.0 INTRODUCTION

# -----

# FROM A USER'S POINT OF VIEW, ORBITAL INTEGRATION IS ESSENTIALLY THE SAME AS THE 278 INTEGRATION PROGRAM. THE SAME ENTRANCES TO THE PROGRAM WILL BE MAINTAINED, THE SAME STALLING ROUTINE WILL BE USED AND OUTPUT WILL STILL BE VIA THE PUSHLIST. THE PRIMARY DIFFERENCES TO A USER INVOLVE THE ADDED CAPABILITY OF TERMINATING INTEGRATION AT A SPECIFIC FINAL RADIUS AND THE DIFFERENCE IN STATE VECTOR SCALING INSIDE AND OUTSIDE THE LUNAR SPHERE OF INFLUENCE.

# IN ORDER TO MAKE THE CSM(LEM)PREC AND CSM(LEM)CONIC ENTRANCES SIMILAR TO FLIGHT 278, THE INTEGRATION PROGRAM WILL ITSELF SET THE FINAL RADIUS (RFINAL) TO 0 SO THAT REACHING THE DESIRED TIME ONLY WILL TERMINATE INTEGRATION. THE DP REGISTER RFINAL MUST BE SET BY USERS OF INTEGRVS AND INTEGRV, AND MUST BE DONE AFTER THE CALL TC INTSTALL.

# WHEN THE LM IS ON THE LUNAR SURFACE (INDICATED BY LUNAR SURFACE FLAG SET) CALLS TO LEMCONIC, LEMPREC, AND INTEGRV WITH VINFLAG = 0 WILL RESULT IN THE USE OF THE PLANETARY INERTIAL ORIENTATION SUBROUTINES TO PROVIDE BOTH THE LM'S POSITION AND VELOCITY IN THE REFERENCE COORDINATE SYSTEM. THE PROGRAM WILL PROVIDE OUTPUT AS IF INTEGRATION WAS USED. THAT IS, THE PUSHLIST WILL BE SET AS NOTED BELOW AND THE PERMANENT STATE VECTOR UPDATED WHEN SPECIFIED BY AN INTEGRV CALL.

# USERS OF INTEGRVS DESIRING INTEGRATION (INTYPFLG = 0) SHOULD NOTE THAT THE OBLATENESS PERTURBATION COMPUTATION IN LUNAR ORBIT IS TIME DEPENDENT. THEREFORE, THE USER SHOULD SUPPLY AN INITIAL STATE VECTOR VALID AT SOME REAL TIME AND THE DESIRED TIME (TDECI) ALSO AT SOME REAL TIME. FOR CONIC "INTEGRATION" THE USER MAY STILL USE ZERO AS THE INITIAL TIME AND DELTA TIME AS THE DESIRED TIME.

## # 2.0 GENERAL DESCRIPTION

# -----

# THE INTEGRATION PROGRAM OPERATES AS A CLOSED INTERPRETIVE SUBROUTINE AND PERFORMS THESE FUNCTIONS --

- # 1) INTEGRATES (PRECISION OR CONIC) EITHER CSM OR LM STATE VECTOR
- # 2) INTEGRATES THE W-MATRIX
- # 3) PERMANENT OR TEMPORARY UPDATE OF THE STATE VECTOR

# THERE ARE SIX ENTRANCES TO THE INTEGRATION PROGRAM. FOUR OF THESE (CSMPREC, LEMPREC, CSMCONIC, LEMCONIC) SET ALL THE FLAGS REQUIRED IN THE INTEGRATION PROGRAM ITSELF TO CAUSE THE PRECISION OR CONIC INTEGRATION (KEPLER) OF THE LM OR CSM STATE VECTOR, AS THE NAMES SUGGEST. ONE ENTRANCE (INTEGRVS) PERMITS THE CALLING PROGRAM TO PROVIDE A STATE VECTOR TO BE INTEGRATED. THE CALLING PROGRAM MUST SET THE FLAGS INDICATING (1) PRECISION OR CONIC INTEGRATION, (2) IN OR OUT OF LUNAR SPHERE, (3) MIDCOURSE OR NOT, AND THE INTEGRATION PROGRAM COMPLETES THE FLAG SETTING TO BYPASS W-MATRIX INTEGRATION. THE LAST ENTRANCE (INTEGRV, USED IN GENERAL BY THE NAVIGATION PROGRAMS) PERMITS THE CALLER TO SET FIVE FLAGS (NOT MOONFLAG OR MIDFLAG) BUT NOT TO INPUT A STATE VECTOR. ANY PROGRAM WHICH CALLS INTEGRVS OR INTEGRV MUST CALL INTSTALL BEFORE IT SETS THE INTEGRATION FLAGS AND/OR STATE VECTOR.

# THREE SETS OF 42 REGISTERS AND 2 FLAGS ARE USED FOR THE STATE VECTORS. TWO SETS, WHICH MAY NOT BE OVERLAYED, ARE USED FOR THE PERMANENT STATE VECTORS FOR THE CSM AND LM. THE THIRD SET, WHICH MAY BE OVERLAYED WHEN INTEGRATION IS NOT BEING DONE, IS USED IN THE COMPUTATIONS.

# THE PERMANENT STATE VECTORS WILL BE PERIODICALLY UPDATED SO THAT THE VECTORS WILL NOT BE OLDER THAN 4 TIMESTEPS. THE PERMANENT STATE VECTORS WILL ALSO BE UPDATED WHENEVER THE W-MATRIX IS INTEGRATED OR WHEN A CALLER OF INTEGRV SETS STATEFLG (THE NAVIGATION PROGRAMS P20, P22.)

#

# APPENDIX B OF THE USERS' GUIDE LISTS THE STATE VECTOR QUANTITIES.

#

## # 2.1 RESTARTS

#

# PHASE CHANGES WILL BE MADE IN THE INTEGRATION PROGRAM ONLY FOR THE INTEGRV ENTRANCE (I.E., WHEN THE W-MATRIX IS  
# INTEGRATED OR PERMANENT STATE VECTOR IS UPDATED.) THE GROUP NUMBER USED WILL BE THAT FOR THE P20-25 PROGRAMS  
# (I.E., GROUP2) SINCE THE INTEGRV ENTRANCE WILL ONLY BE USED BY THESE PROGRAMS. IF A RESTART OCCURS DURING AN  
# INTEGRATION OF THE STATE VECTOR ONLY, THE RECOVERY WILL BE TO THE LAST PHASE IN THE CALLING PROGRAM. CALLING  
# PROGRAMS WHICH USE THE INTEGRV OR INTEGRVS ENTRANCE OF INTEGRATION SHOULD ENSURE THAT IF PHASE CHANGING IS DONE  
# THAT IT IS PRIOR TO SETTING THE INTEGRATION INPUTS IN THE PUSHLIST.  
# THIS IS BECAUSE THE PUSHLIST IS LOST DURING A RESTART.

#

## # 2.2 SCALING

#

# THE INTEGRATION ROUTINE WILL MAINTAIN THE PERMANENT MEMORY STATE VECTORS IN THE SCALING AND UNITS DEFINED IN  
# APPENDIX B OF THE USERS GUIDE. THE SCALING OF THE OUTPUT POSITION VECTOR DEPENDS ON THE ORIGIN OF THE COORDINATE  
# SYSTEM AT THE DESIRED INTEGRATION TIME. THE COORDINATE SYSTEM TRANSFORMATION WILL BE DONE AUTOMATICALLY ON  
# MULTIPLE TIMESTEP ENCKE INTEGRATION ONLY. THUS IT IS POSSIBLE TO HAVE OUTPUT FROM SUCCESSIVE INTEGRATIONS IN  
# DIFFERENT SCALING.  
# HOWEVER, RATT, VATT WILL ALWAYS BE SCALED THE SAME.

#

## # 3.0 INPUT/OUTPUT

# -----

#

# PROGRAM INPUTS ARE THE FLAGS DESCRIBED IN APPENDIX A AND THE PERMANENT STATE VECTOR QUANTITIES DESCRIBED IN  
# APPENDIX B OF THE USERS GUIDE, PLUS THE DESIRED TIME TO INTEGRATE TO IN TDEC1 (A PUSH LIST LOCATION).  
# FOR INTEGRVS, THE RCV,VCV,TET OR THE TEMPORARY STATE VECTOR MUST BE SET, PLUS MOONFLAG AND MIDFLAG

#

# FOR SIMULATION THE FOLLOWING QUANTITIES MUST BE PRESET ---

#

|               |                           |        | EARTH | MOON |
|---------------|---------------------------|--------|-------|------|
|               |                           |        | 29    | 27   |
| RRECTCSM(LEM) | RECTIFIED POSITION VECTOR | METERS | 2     | 2    |

#

|               |                           |        |        |        |
|---------------|---------------------------|--------|--------|--------|
| VRECTCSM(LEM) | RECTIFIED VELOCITY VECTOR | M/CSEC | 7<br>2 | 5<br>2 |
|---------------|---------------------------|--------|--------|--------|

#

|             |                            |      |         |         |
|-------------|----------------------------|------|---------|---------|
| TETCSM(LEM) | TIME STATE VECTOR IS VALID | CSEC | 28<br>2 | 28<br>2 |
|-------------|----------------------------|------|---------|---------|

CUSTOMARILY 0, BUT NOTE LUNAR ORBIT DEPENDENCE ON REAL TIME.

#

|                |                    |        |         |         |
|----------------|--------------------|--------|---------|---------|
| DELTAVCSM(LEM) | POSITION DEVIATION | METERS | 22<br>2 | 18<br>2 |
|----------------|--------------------|--------|---------|---------|

0 IF TCCSM(LEM) = 0

#

|             |                    |        |        |         |
|-------------|--------------------|--------|--------|---------|
| NUVCSM(LEM) | VELOCITY DEVIATION | M/CSEC | 3<br>2 | -1<br>2 |
|-------------|--------------------|--------|--------|---------|

0 IF TCCSM(LEM) = 0

#

# INTEGRATION\_INITIALIZATION

|    |   |                                                                        |                          |          |                               |    |      |        |     |    |    |    |
|----|---|------------------------------------------------------------------------|--------------------------|----------|-------------------------------|----|------|--------|-----|----|----|----|
| 1  |   |                                                                        |                          |          |                               |    |      |        |     |    | 1  |    |
| 2  | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 2  |
| 3  | # | RCVSM(LEM)                                                             |                          |          | CONIC POSITION                |    |      | METERS | 29  | 27 | 3  |    |
| 4  | # | EQUALS RRECTCSM(LEM) IF                                                |                          |          |                               |    |      |        |     |    |    | 4  |
| 5  | # | TCCSM(LEM) = 0                                                         |                          |          |                               |    |      |        |     |    |    | 5  |
| 6  | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 6  |
| 7  | # | VCVCSM(LEM)                                                            |                          |          | CONIC VELOCITY                |    |      | M/CSEC | 7   | 5  | 7  |    |
| 8  | # | EQUALS VRECTCSM(LEM) IF                                                |                          |          |                               |    |      |        |     |    |    | 8  |
| 9  | # | TCCSM(LEM) = 0                                                         |                          |          |                               |    |      |        |     |    |    | 9  |
| 10 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 10 |
| 11 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 11 |
| 12 | # | TCCSM(LEM)                                                             |                          |          | TIME SINCE RECTIFICATION      |    |      | CSECS  | 28  | 28 | 12 |    |
| 13 | # | CUSTOMARILY 0                                                          |                          |          |                               |    |      |        |     |    |    | 13 |
| 14 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 14 |
| 15 | # | XKEPCSM(LEM)                                                           |                          |          | ROOT OF KEPLER'S EQUATION     |    |      | 1/2    | 17  | 16 | 15 |    |
| 16 | # | 0 IF TCCSM(LEM) = 0                                                    |                          |          |                               |    |      |        |     |    |    | 16 |
| 17 | # | CMOONFLG                                                               |                          |          | PERMANENT FLAGS CORRESPONDING |    |      | 0      | 0   |    | 17 |    |
| 18 | # | CMIDFLAG                                                               |                          |          | TO MOONFLAG AND MIDFLAG       |    |      | 0,1    | 0,1 |    | 18 |    |
| 19 | # | LMOONFLG                                                               |                          |          | C = CSM, L = LM               |    |      | 0      | 0   |    | 19 |    |
| 20 | # | LMIDFLAG                                                               |                          |          |                               |    |      | 0,1    | 0,1 |    | 20 |    |
| 21 | # | SURFFLAG                                                               |                          |          | LUNAR SURFACE FLAG            |    |      | 0,1    | 0,1 |    | 21 |    |
| 22 | # | IN ADDITION, IF (L)CMIDFLAG IS SET, THE INITIAL INPUT VALUES FOR LUNAR |                          |          |                               |    |      |        |     |    |    | 22 |
| 23 | # | SOLAR EPHEMERIDES SUBROUTINE AND PLANETARY INERTIAL ORIENTATION SUB-   |                          |          |                               |    |      |        |     |    |    | 23 |
| 24 | # | ROUTINE MUST BE PRESET.                                                |                          |          |                               |    |      |        |     |    |    | 24 |
| 25 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 25 |
| 26 | # | OUTPUT                                                                 |                          |          |                               |    |      |        |     |    |    | 26 |
| 27 | # | AFTER EVERY CALL TO INTEGRATION                                        |                          |          |                               |    |      |        |     |    |    | 27 |
| 28 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 28 |
| 29 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 29 |
| 30 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 30 |
| 31 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 31 |
| 32 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 32 |
| 33 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 33 |
| 34 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 34 |
| 35 | # | 0D                                                                     | RATT                     | POSITION | METERS                        | 29 | 29   |        |     |    | 35 |    |
| 36 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 36 |
| 37 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 37 |
| 38 | # | 6D                                                                     | VATT                     | VELOCITY | M/CSEC                        | 7  | 7    |        |     |    | 38 |    |
| 39 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 39 |
| 40 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 40 |
| 41 | # | 12D                                                                    | TAT                      | TIME     |                               | 28 | 28   |        |     |    | 41 |    |
| 42 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 42 |
| 43 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 43 |
| 44 | # | 14D                                                                    | RATT1                    | POSITION | METERS                        | 29 | 27   |        |     |    | 44 |    |
| 45 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 45 |
| 46 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 46 |
| 47 | # | 20D                                                                    | VATT1                    | VELOCITY | M/CSEC                        | 7  | 5    |        |     |    | 47 |    |
| 48 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 48 |
| 49 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 49 |
| 50 | # | 26D                                                                    | MU(P)                    | MU       | 3 2                           | 36 | 30   |        |     |    | 50 |    |
| 51 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 51 |
| 52 | # | X1                                                                     | MUTABLE ENTRY            |          |                               | -2 | -10D |        |     |    | 52 |    |
| 53 | # |                                                                        |                          |          |                               |    |      |        |     |    |    | 53 |
| 54 | # | X2                                                                     | COORDINT                 |          |                               |    |      |        |     |    | 54 |    |
| 55 | # | X2                                                                     | COORDINATE SYSTEM ORIGIN |          |                               | 0  | 2    |        |     |    | 55 |    |
| 56 | # | (THIS, NOT MOONFLAG, SHOULD BE                                         |                          |          |                               |    |      |        |     |    |    | 56 |
| 57 |   |                                                                        |                          |          |                               |    |      |        |     |    |    | 57 |
| 58 |   |                                                                        |                          |          |                               |    |      |        |     |    |    | 58 |
| 59 |   |                                                                        |                          |          |                               |    |      |        |     |    |    | 59 |
| 60 |   |                                                                        |                          |          |                               |    |      |        |     |    |    | 60 |

1412THE



```
1 #
2 # USED TO DETERMINE ORIGIN.)
3 #
4 # IN ADDITION TO THE ABOVE, THE PERMANENT STATE VECTOR IS UPDATED WHENEVER
5 # STATEFLG WAS SET AND WHENEVER A W-MATRIX IS TO BE INTEGRATED. THE PUSH
6 # COUNTER IS SET TO 0 AND OVERFLOW IS CLEARED BEFORE RETURNING TO THE
7 # CALLING PROGRAM.
8 #
9 # 4.0 CALLING SEQUENCES AND SAMPLE CODE
10 # -----
11 #
12 # A) PRECISION ORBITAL INTEGRATION. CSMPREC, LEMPREC ENTRANCES
13 # L-X STORE TIME TO 96T5791T5 T 95 PUS L9ST (T4531)
14 # L CALL
15 # L+1 CSMPREC (OR LEMPREC)
16 # L+2 RETURN
17 # INPUT
18 # TDEC1 (PD 32D) TIME TO INTEGRATE TO...CENTISECONDS SCALED 2
19 # OUTPUT
20 # THE DATA LISTED IN SECTION 3.0 PLUS
21 # RQVV POSITION VECTOR OF VEHICLE WITH RESPECT TO SECONDARY
22 # BODY... METERS B-29 ONLY IF MIDFLAG = DIMOFLAG = 1
23 # B) CONIC INTEGRATION. CSMCONIC, LEMCONIC ENTRANCES
24 # L-X STORE TIME IN PUSH LIST (TDEC1)
25 # L CALL
26 # L+1 CSMCONIC (OR LEMCONIC)
27 # INPUT/OUTPUT
28 # SAME AS PRECISION INTEGRATION, EXCEPT RQVV NOT SET
29 # C) INTEGRATE GIVEN STATE VECTOR. INTEGRVS ENTRANCE
30 # CALL
31 # INTSTALL
32 # VLOAD
33 # POSITION VECTOR
34 # STOVL RCV
35 # VELOCITY VECTOR
36 # STODL VCV
37 # TIME STATE VECTOR VALID
38 # STODL TET
39 # FINAL RADIUS
40 # STORE RFINAL
41 # SET(CLEAR) SET(CLEAR)
42 # INTYPFLAG
43 # MOONFLAG
44 # SET(CLEAR) DLOAD
45 # DESIRED TIME
46 # STCALL TDEC1
47 # INTEGRVS
48 # INPUT
49 # RCV POSITION VECTOR METERS
50 # VCV VELOCITY VECTOR M/CSEC
51 # TET TIME OF STATE VECTOR (MAY = 0) CSEC B-28
```

```
1 #
2 # TDEC1 TIME TO INTEGRATE TO CSEC B-28 (PD 32D)
3 # (MAY BE INCREMENT IF TET=0)
4 # OUTPUT
5 # SAME AS FOR PRECISION OR CONIC INTEGRATION,
6 # DEPENDING ON INTYPFLG.
7 # D) INTEGRATE STATE VECTOR. INTGRV ENTRANCE
8 # L-X STORE TIME IN PUSH LIST (TDEC1) (MAY BE DONE AFTER CALL TO INTSTALL)
9 # L-8 CALL
10 # L-7
11 # L-6 SET(CLEAR) SET(CLEAR)
12 # L-5 VINTFLAG 1=CSM, 0=LM
13 # L-4 INTYPFLAG 1=CONIC, 0=PRECISION
14 # L-3 SET(CLEAR) SET(CLEAR)
15 # L-2 DIMOFLAG 1=W-MATRIX, 0=NO W-MATRIX
16 # L-1 D6OR9FLG 1=9X9, 0=6X6
17 # L SET DLOAD
18 # L+1 STATEFLG DESIRE PERMANENT UPDATE
19 # L+2 FINAL RAD. OF STATE VECTOR
20 # L+3 STCALL RFINAL
21 # L+4 INTEGRV
22 # L CALL NORMAL USE -- WILL UPDATE STATE
23 # L+1 INTEGRV VECTOR IF DIMOFLAG=1. (STATEFLG IS
24 # L+2 RETURN ALWAYS RESET IN INTEGRATION AFTER
25 # IT USED.)
26 # INPUT
27 # TDEC1 (PD 32D) TIME TO INTEGRATE TO CSEC B-28
28 # OUTPUT
29 # SAME AS FOR PRECISION OR CONIC INTEGRATION
30 # THE PROGRAM WILL SET MOONFLAG, MIDFLAG DEPENDING ON
31 # THE PERMANENT STATE VECTOR REPRESENTATION.
32 #
33 # BANK 11
34 # SETLOC INTINIT
35 # BANK
36 # EBANK= RRECTCSM
37 # STATEINT COUNT* $$/INTIN
38 # TC PHASCHNG
39 # OCT 00052
40 # CAF PRI05
41 # TC FINDVAC
42 # EBANK= RRECTCSM
43 # 2CADR STATINT1
44 #
45 # TC TASKOVER
46 # STATINT1 TC INTPRET
47 # BON RTB
48 # QUITFLAG # KILL INTEGRATION UNTIL NEXT P00.
49 # NOINT
50 # LOADTIME
51 # STORE TDEC1
```



|    |                                                                    |          |          |    |
|----|--------------------------------------------------------------------|----------|----------|----|
| 1  |                                                                    |          |          | 1  |
| 2  | CALL                                                               |          |          | 2  |
| 3  |                                                                    | INTSTALL |          | 3  |
| 4  |                                                                    | SET      | CALL     | 4  |
| 5  |                                                                    |          | NODOFLAG | 5  |
| 6  |                                                                    |          | SETIFLGS | 6  |
| 7  | GOTO                                                               |          |          | 7  |
| 8  |                                                                    |          | STATEUP  | 8  |
| 9  | 600SECS                                                            | 2DEC     | 60000    | 9  |
| 10 |                                                                    |          |          | 10 |
| 11 | ENDINT                                                             | CLEAR    | EXIT     | 11 |
| 12 |                                                                    |          | STATEFLG | 12 |
| 13 |                                                                    | TC       | PHASCHNG | 13 |
| 14 |                                                                    | OCT      | 20032    | 14 |
| 15 |                                                                    | EXTEND   |          | 15 |
| 16 |                                                                    | DCA      | 600SECS  | 16 |
| 17 |                                                                    | TC       | LONGCALL | 17 |
| 18 |                                                                    | EBANK=   | RRECTHIS | 18 |
| 19 |                                                                    | 2CADR    | STATEINT | 19 |
| 20 |                                                                    |          |          | 20 |
| 21 |                                                                    | TC       | ENDOFJOB | 21 |
| 22 | SETIFLGS                                                           | SET      | CLEAR    | 22 |
| 23 |                                                                    |          | STATEFLG | 23 |
| 24 |                                                                    |          | INTYPFLG | 24 |
| 25 |                                                                    | CLEAR    | CLEAR    | 25 |
| 26 |                                                                    |          | DIM0FLAG | 26 |
| 27 |                                                                    |          | D6OR9FLG | 27 |
| 28 |                                                                    |          |          | 28 |
| 29 | NOINT                                                              | RVQ      |          | 29 |
| 30 |                                                                    | EXIT     |          | 30 |
| 31 |                                                                    | TC       | PHASCHNG | 31 |
| 32 |                                                                    | OCT      | 00002    | 32 |
| 33 |                                                                    | TC       | DOWNFLAG | 33 |
| 34 |                                                                    | ADRES    | QUITFLAG | 34 |
| 35 |                                                                    | TC       | ENDOFJOB | 35 |
| 36 |                                                                    |          |          | 36 |
| 37 | # ATOPCSM TRANSFERS RRECT TO RRECT +41 TO RRECTCSM TO RRECTCSM +41 |          |          | 37 |
| 38 | #                                                                  |          |          | 38 |
| 39 | # CALLING SEQUENCE                                                 |          |          | 39 |
| 40 | #                                                                  | L        | CALL     | 40 |
| 41 | #                                                                  | L+1      | ATOPCSM  | 41 |
| 42 | #                                                                  |          |          | 42 |
| 43 | # NORMAL EXIT AT L+2                                               |          |          | 43 |
| 44 |                                                                    |          |          | 44 |
| 45 | ATOPCSM                                                            | STQ      | RTB      | 45 |
| 46 |                                                                    |          | S2       | 46 |
| 47 |                                                                    |          | MOVEACSM | 47 |
| 48 |                                                                    | SET      | CALL     | 48 |
| 49 |                                                                    |          |          | 49 |
| 50 |                                                                    |          | CMOONFLG | 50 |
| 51 |                                                                    | BON      | SVDWN1   | 51 |
| 52 |                                                                    |          | CLRGO    | 52 |
| 53 |                                                                    |          |          | 53 |
| 54 |                                                                    |          |          | 54 |
| 55 |                                                                    |          |          | 55 |
| 56 |                                                                    |          |          | 56 |
| 57 |                                                                    |          |          | 57 |
| 58 |                                                                    |          |          | 58 |
| 59 |                                                                    |          |          | 59 |
| 60 |                                                                    |          |          | 60 |

```
1
2 MOONFLAG
3 S2
4 CMOONFLG
5 S2
6 MOVEACSM TC SETBANK
7 TS DIFEQCNT # INITIALIZE INDEX
8 INDEX DIFEQCNT
9 CA RRECT
10 INDEX DIFEQCNT
11 TS RRECTCSM
12 CCS DIFEQCNT # IS TRANSFER COMPLETE
13 TCF MOVEACSM +1 # NO-LOOP
14 TC DANZIG # COMPLETE -- RETURN
15
16 # PTOACSM TRANSFERS RRECTCSM TO RRECTCSM +41 TO RRECT TO RRECT +41
17 #
18 # CALLING SEQUENCE
19 # L CALL PTOACSM
20 #
21 #
22 # NORMAL EXIT AT L+2
23
24 PTOACSM RTB BON
25 MOVEPCSM
26 CMOONFLG
27 SETMOON
28 CLRMOON CLEAR SSP
29 MOONFLAG
30 PBODY
31 0
32
33 SETMOON RVQ
34 SET SSP
35 MOONFLAG
36 PBODY
37 2
38
39 MOVEPCSM RVQ
40 TC SETBANK
41 TS DIFEQCNT
42 INDEX DIFEQCNT
43 CA RRECTCSM
44 INDEX DIFEQCNT
45 TS RRECT
46 CCS DIFEQCNT
47 TCF MOVEPCSM +1
48 TC DANZIG
49
50 # ATOPLEM TRANSFERS RRECT TO RRECT +41 TO RRECTLEM TO RRECTLEM +41
51 ATOPLEM STQ RTB
```

| 142THE |  |
|--------|--|
| 1      |  |
| 2      |  |
| 3      |  |
| 4      |  |
| 5      |  |
| 6      |  |
| 7      |  |
| 8      |  |
| 9      |  |
| 10     |  |
| 11     |  |
| 12     |  |
| 13     |  |
| 14     |  |
| 15     |  |
| 16     |  |
| 17     |  |
| 18     |  |
| 19     |  |
| 20     |  |
| 21     |  |
| 22     |  |
| 23     |  |
| 24     |  |
| 25     |  |
| 26     |  |
| 27     |  |
| 28     |  |
| 29     |  |
| 30     |  |
| 31     |  |
| 32     |  |
| 33     |  |
| 34     |  |
| 35     |  |
| 36     |  |
| 37     |  |
| 38     |  |
| 39     |  |
| 40     |  |
| 41     |  |
| 42     |  |
| 43     |  |
| 44     |  |
| 45     |  |
| 46     |  |
| 47     |  |
| 48     |  |
| 49     |  |
| 50     |  |
| 51     |  |
| 52     |  |
| 53     |  |
| 54     |  |
| 55     |  |
| 56     |  |
| 57     |  |
| 58     |  |
| 59     |  |
| 60     |  |
| 61     |  |
| 62     |  |
| 63     |  |
| 64     |  |
| 65     |  |
| 66     |  |
| 67     |  |
| 68     |  |
| 69     |  |
| 70     |  |
| 71     |  |
| 72     |  |
| 73     |  |
| 74     |  |
| 75     |  |
| 76     |  |
| 77     |  |
| 78     |  |
| 79     |  |
| 80     |  |

```
1
2 RP-TO-R
3 STOVL RCV
4 ZUNIT
5 STODL OD
6 TET
7 STODL 6D
8 5/8
9 SET CALL # NEEDED FOR SETTING X1 ON EXIT
10 MOONFLAG
11 RP-TO-R
12 VXV VXSC
13 RCV
14 OMEGMOON
15 STOVL VCV
16 ZEROVEC
17 STORE TDELTA
18 AXT,2 SXA,2
19 2
20 PBODY
21 STCALL TNUV
22 A-PCHK
23 SETBANK CAF INTBANK
24 TS BBANK
25 CAF FORTYONE
26 TC Q
27 EBANK= RRECTCSM
28 INTBANK BBCON INTEGRV
29
30 # SPECIAL PURPOSE ENTRIES TO ORBITAL INTEGRATION. THESE ROUTINES PROVIDE ENTRANCES TO INTEGRATION WITH
31 # APPROPRIATE SWITCHES SET OR CLEARED FOR THE DESIRED INTEGRATION.
32 #
33 # CSMPREC AND LEMPREC PERFORM ORBIT INTEGRATION BY THE ENCKE METHOD TO THE TIME INDICATED IN TDEC1.
34 # ACCELERATIONS DUE TO OBLATENESS ARE INCLUDED. NO W-MATRIX INT. IS DONE.
35 # THE PERMANENT STATE VECTOR IS NOT UPDATED.
36 #
37 # CSMCONIC AND LEMCONIC PERFORM ORBIT INTEG. BY KEPLER'S METHOD TO THE TIME INDICATED IN TDEC1.
38 # NO DISTURBING ACCELERATIONS ARE INCLUDED. IN THE PROGRAM FLOW THE GIVEN
39 # STATE VECTOR IS RECTIFIED BEFORE SOLUTION OF KEPLER'S EQUATION.
40 #
41 # THE ROUTINES ASSUME THAT THE CSM (LEM) STATE VECTOR IN P-MEM IS VALID.
42 # SWITCHES SET PRIOR TO ENTRY TO THE MAIN INTEG. PROG ARE AS FOLLOWS:
43 # CSMPREC CSMCONIC LEMPREC LEMCONIC
44 # VINTFLAG SET SET CLEAR CLEAR
45 # INTYPFLG CLEAR SET CLEAR SET
46 # DIMOFLAG CLEAR CLEAR CLEAR CLEAR
47 #
48 # CALLING SEQUENCE
49 # L-X STORE TDEC1
50 # L CALL (STCALL TDEC1)
```

# INTEGRATION\_INITIALIZATION

|    |          |                                          |                                       |    |
|----|----------|------------------------------------------|---------------------------------------|----|
| 1  |          |                                          |                                       | 1  |
| 2  | #        | L+1                                      | CSMPREC (CSMCONIC, LEMPREC, LEMCONIC) | 2  |
| 3  | #        |                                          |                                       | 3  |
| 4  | #        | NORMAL EXIT TO L+2                       |                                       | 4  |
| 5  | #        |                                          |                                       | 5  |
| 6  | #        | SUBROUTINES CALLED                       |                                       | 6  |
| 7  | #        | INTEGRV1                                 |                                       | 7  |
| 8  | #        | PRECOUT FOR CSMPREC AND LEMPREC          |                                       | 8  |
| 9  | #        | CONICOUT FOR CSMCONIC AND LEMCONIC       |                                       | 9  |
| 10 | #        |                                          |                                       | 10 |
| 11 | #        | OUTPUT -- SEE PAGE 2 OF THIS LOG SECTION |                                       | 11 |
| 12 | #        |                                          |                                       | 12 |
| 13 | #        | INPUT                                    |                                       | 13 |
| 14 | #        | TDEC1                                    | TIME TO INTEGRATE TO. CSECS B-28      | 14 |
| 15 |          |                                          |                                       | 15 |
| 16 | CSMPREC  | STQ                                      | CALL                                  | 16 |
| 17 |          |                                          | X1                                    | 17 |
| 18 |          |                                          | INTSTALL                              | 18 |
| 19 |          | SXA,1                                    | SET                                   | 19 |
| 20 |          |                                          | IRETURN                               | 20 |
| 21 |          |                                          | VINTFLAG                              | 21 |
| 22 |          |                                          |                                       | 22 |
| 23 | IFLAGP   | SET                                      | CLEAR                                 | 23 |
| 24 |          |                                          | PRECIFLG                              | 24 |
| 25 |          |                                          | DIMOFLAG                              | 25 |
| 26 |          | CLRGO                                    |                                       | 26 |
| 27 |          |                                          | INTYPFLG                              | 27 |
| 28 |          |                                          | INTEGRV1                              | 28 |
| 29 | LEMPREC  | STQ                                      | CALL                                  | 29 |
| 30 |          |                                          | X1                                    | 30 |
| 31 |          |                                          | INTSTALL                              | 31 |
| 32 |          | SXA,1                                    | CLRGO                                 | 32 |
| 33 |          |                                          | IRETURN                               | 33 |
| 34 |          |                                          | VINTFLAG                              | 34 |
| 35 |          |                                          | IFLAGP                                | 35 |
| 36 |          |                                          |                                       | 36 |
| 37 | CSMCONIC | STQ                                      | CALL                                  | 37 |
| 38 |          |                                          | X1                                    | 38 |
| 39 |          |                                          | INTSTALL                              | 39 |
| 40 |          | SXA,1                                    | SET                                   | 40 |
| 41 |          |                                          | IRETURN                               | 41 |
| 42 |          |                                          | VINTFLAG                              | 42 |
| 43 | IFLAGC   | CLEAR                                    | SETGO                                 | 43 |
| 44 |          |                                          | DIMOFLAG                              | 44 |
| 45 |          |                                          | INTYPFLG                              | 45 |
| 46 |          |                                          | INTEGRV1                              | 46 |
| 47 | LEMCONIC | STQ                                      | CALL                                  | 47 |
| 48 |          |                                          | X1                                    | 48 |
| 49 |          |                                          | INTSTALL                              | 49 |
| 50 |          | SXA,1                                    | CLRGO                                 | 50 |
| 51 |          |                                          | IRETURN                               | 51 |
| 52 |          |                                          |                                       | 52 |
| 53 |          |                                          |                                       | 53 |
| 54 |          |                                          |                                       | 54 |
| 55 |          |                                          |                                       | 55 |
| 56 |          |                                          |                                       | 56 |
| 57 |          |                                          |                                       | 57 |
| 58 |          |                                          |                                       | 58 |
| 59 |          |                                          |                                       | 59 |
| 60 |          |                                          |                                       | 60 |

VINTFLAG  
IFLAGC

INTEGRVS            SET        SSP  
PRECIFLG

BOF        SSP  
MOONFLAG  
+3  
PBODY

STQ        VLOAD  
IRETURN

STORE        ZEROVEC  
STCALL        TDELTA  
TNUV

CLEAR        RECTIFY  
SET  
DIMOFLAG

SETGO        NEWIFLG  
RPQFLAG  
ALOADED

# INTEGRV IS AN ENTRY TO ORBIT INTEGRATION WHICH PERMITS THE CALLER,  
# NORMALLY THE NAVIGATION PROGRAM, TO SET THE INTEG. FLAGS. THE ROUTINE  
# IS ENTERED AT INTEGRV1 BY CSMPREC ET AL. AND AT ALOADED BY INTEGRVS.  
# THE ROUTINE SETS UP A-MEMORY IF ENTERED AT INTEGRV,1 AND SETS THE INTEG.  
# PROGRAM FOR PRECISION OR CONIC.

# THE CALLER MUST FIRST CALL INTSTALL TO CHECK IF INTEG. IS IN USE BEFORE  
# SETTING ANY FLAGS.

# THE FLAGS WHICH SHOULD BE SET OR CLEARED ARE  
#        VINTFLAG            (IGNORED WHEN ENTERED FROM INTEGRVS)  
#        INTYPFLG  
#        DIMOFLAG  
#        D6OR9FLG

# CALLING SEQUENCE  
#        L-X        CALL  
#        L-Y            INTSTALL  
#        L-1        SET OR CLEAR ALL FOUR FLAGS. ALSO CAN SET STATEFLG IF DESIRED  
#                    AND DIMOFLAG IS CLEAR.  
#        L        CALL  
#        L+1            INTEGRV

# INITIALIZATION  
#        FLAGS AS ABOVE  
#        STORE TIME TO INTEGRATE TO IN TDEC1

# OUTPUT  
#        RATT        AS  
#        VATT            DEFINED



# INTEGRATION\_INITIALIZATION

|    |          |       |          |                                     |                        |    |
|----|----------|-------|----------|-------------------------------------|------------------------|----|
| 1  |          |       |          |                                     | 1                      |    |
| 2  | #        | TAT   | BEFORE   |                                     | 2                      |    |
| 3  |          |       |          |                                     | 3                      |    |
| 4  | INTEGRV  | STQ   |          |                                     | 4                      |    |
| 5  | INTEGRV1 | SET   | IRETURN  |                                     | 5                      |    |
| 6  |          |       | SET      |                                     | 6                      |    |
| 7  | INTEGRV2 | SSP   | RPQFLAG  |                                     | 7                      |    |
| 8  |          |       | NEWIFLG  |                                     | 8                      |    |
| 9  |          |       |          |                                     | 9                      |    |
| 10 |          | BON   | QPRET    |                                     | 10                     |    |
| 11 |          |       | ALOADED  |                                     | 11                     |    |
| 12 |          |       | GOTO     |                                     | 12                     |    |
| 13 |          |       | VINTFLAG |                                     | 13                     |    |
| 14 |          |       | PTOACSM  |                                     | 14                     |    |
| 15 |          |       | PTOALEM  |                                     | 15                     |    |
| 16 | ALOADED  | DLOAD |          |                                     | 16                     |    |
| 17 |          | STORE | TDEC1    |                                     | 17                     |    |
| 18 |          |       | TDEC     |                                     | 18                     |    |
| 19 |          |       | GOTO     |                                     | 19                     |    |
| 20 |          | BOFF  | INTYPFLG |                                     | 20                     |    |
| 21 |          |       | TESTLOOP |                                     | 21                     |    |
| 22 |          |       |          |                                     | 22                     |    |
| 23 | A-PCHK   | BOF   | RVCON    |                                     | 23                     |    |
| 24 |          |       | EXIT     |                                     | 24                     |    |
| 25 |          |       | STATEFLG |                                     | 25                     |    |
| 26 |          | TC    | RECTOUT  |                                     | 26                     |    |
| 27 |          |       | PHASCHNG |                                     | 27                     |    |
| 28 |          |       | 04022    |                                     | 28                     |    |
| 29 |          | TC    | UPFLAG   | # PHASE CHANGE HAS OCCURRED BETWEEN | 29                     |    |
| 30 |          |       | ADRES    | REINTFLG                            | # INTSTALL AND INTWAKE | 30 |
| 31 |          |       | TC       | INTPRET                             |                        | 31 |
| 32 |          | SSP   |          |                                     | 32                     |    |
| 33 |          |       | QPRET    |                                     | 33                     |    |
| 34 |          |       | PHEXIT   |                                     | 34                     |    |
| 35 |          | BON   | GOTO     |                                     | 35                     |    |
| 36 |          |       | VINTFLAG |                                     | 36                     |    |
| 37 |          |       | ATOPCSM  |                                     | 37                     |    |
| 38 | PHEXIT   | CALL  | ATOPLEM  |                                     | 38                     |    |
| 39 | RECTOUT  | SETPD | GRP2PC   |                                     | 39                     |    |
| 40 |          |       | CALL     |                                     | 40                     |    |
| 41 |          |       | 0        |                                     | 41                     |    |
| 42 |          | VLOAD | RECTIFY  |                                     | 42                     |    |
| 43 |          |       | VSL*     |                                     | 43                     |    |
| 44 |          |       | RRECT    |                                     | 44                     |    |
| 45 |          | PDVL  | 0,2      |                                     | 45                     |    |
| 46 |          |       | VSL*     | # RATT TO PDO                       | 46                     |    |
| 47 |          |       | VRECT    |                                     | 47                     |    |
| 48 |          | PDDL  | 0,2      |                                     | 48                     |    |
| 49 |          |       | PDVL     | # VATT TO PD6 TAT TO PD12           | 49                     |    |
| 50 |          |       | TET      |                                     | 50                     |    |
| 51 |          |       |          |                                     | 51                     |    |
| 52 |          |       |          |                                     | 52                     |    |
| 53 |          |       |          |                                     | 53                     |    |
| 54 |          |       |          |                                     | 54                     |    |
| 55 |          |       |          |                                     | 55                     |    |
| 56 |          |       |          |                                     | 56                     |    |
| 57 |          |       |          |                                     | 57                     |    |
| 58 |          |       |          |                                     | 58                     |    |
| 59 |          |       |          |                                     | 59                     |    |
| 60 |          |       |          |                                     | 60                     |    |

|    |  |
|----|--|
| 1  |  |
| 2  |  |
| 3  |  |
| 4  |  |
| 5  |  |
| 6  |  |
| 7  |  |
| 8  |  |
| 9  |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |
| 21 |  |
| 22 |  |
| 23 |  |
| 24 |  |
| 25 |  |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 |  |
| 31 |  |
| 32 |  |
| 33 |  |
| 34 |  |
| 35 |  |
| 36 |  |
| 37 |  |
| 38 |  |
| 39 |  |
| 40 |  |
| 41 |  |
| 42 |  |
| 43 |  |
| 44 |  |
| 45 |  |
| 46 |  |
| 47 |  |
| 48 |  |
| 49 |  |
| 50 |  |
| 51 |  |
| 52 |  |
| 53 |  |
| 54 |  |
| 55 |  |
| 56 |  |
| 57 |  |
| 58 |  |
| 59 |  |
| 60 |  |
| 61 |  |
| 62 |  |
| 63 |  |
| 64 |  |
| 65 |  |
| 66 |  |
| 67 |  |
| 68 |  |
| 69 |  |
| 70 |  |
| 71 |  |
| 72 |  |
| 73 |  |
| 74 |  |
| 75 |  |
| 76 |  |
| 77 |  |
| 78 |  |
| 79 |  |
| 80 |  |

|    |                             |       |                                        |    |
|----|-----------------------------|-------|----------------------------------------|----|
| 1  | # INTEGRATION INTERRELATION |       |                                        | 1  |
| 2  | TESTLOOP                    | BOF   | CLRGO                                  | 2  |
| 3  |                             |       | QUITFLAG                               | 3  |
| 4  |                             |       | +3                                     | 4  |
| 5  |                             |       | STATEFLG                               | 5  |
| 6  |                             |       | INTEXIT                                | 6  |
| 7  | +3                          | SETPD | LXA,2                                  | 7  |
| 8  |                             |       | 10D                                    | 8  |
| 9  |                             |       | PBODY                                  | 9  |
| 10 |                             | VLOAD | ABVAL                                  | 10 |
| 11 |                             |       | RCV                                    | 11 |
| 12 |                             | PUSH  | CLEAR                                  | 12 |
| 13 |                             |       | # RC TO 10D                            | 13 |
| 14 |                             | DSU*  | MIDFLAG                                | 14 |
| 15 |                             |       | BMN                                    | 15 |
| 16 |                             |       | RME,2                                  | 16 |
| 17 |                             |       | # MIDFLAG=0 IF R G.T. RMP              | 17 |
| 18 |                             |       | +3                                     | 18 |
| 19 |                             | SET   |                                        | 19 |
| 20 | NORFINAL                    | DLOAD | MIDFLAG                                | 20 |
| 21 |                             |       | DMP                                    | 21 |
| 22 |                             |       | 10D                                    | 22 |
| 23 |                             |       | 34D                                    | 23 |
| 24 |                             | SR1R  | DDV*                                   | 24 |
| 25 |                             |       | MUEARTH,2                              | 25 |
| 26 |                             | SQRT  | DMP                                    | 26 |
| 27 |                             |       | .3D                                    | 27 |
| 28 |                             | SR3   | SR4                                    | 28 |
| 29 |                             |       | # DT IS TRUNCATED TO A MULTIPLE        | 29 |
| 30 |                             | DLOAD | SL                                     | 30 |
| 31 |                             |       | MPAC                                   | 31 |
| 32 |                             |       | 15D                                    | 32 |
| 33 |                             |       | # OF 128 CSECS.                        | 33 |
| 34 |                             | PUSH  | BOV                                    | 34 |
| 35 |                             |       | MAXDT                                  | 35 |
| 36 |                             | BDSU  | BMN                                    | 36 |
| 37 |                             |       | DT/2MAX                                | 37 |
| 38 | DT/2COMP                    | DLOAD | MAXDT                                  | 38 |
| 39 |                             |       | DSU                                    | 39 |
| 40 |                             |       | TDEC                                   | 40 |
| 41 |                             | RTB   | TET                                    | 41 |
| 42 |                             |       | SL                                     | 42 |
| 43 |                             |       | SGNAGREE                               | 43 |
| 44 |                             |       | 8D                                     | 44 |
| 45 |                             | STORE | DT/2                                   | 45 |
| 46 |                             | BOV   | ABS                                    | 46 |
| 47 |                             |       | # B-19                                 | 47 |
| 48 |                             | DSU   | GETMAXDT                               | 48 |
| 49 |                             |       | BMN                                    | 49 |
| 50 |                             |       | # IS TIME TO INTEG. TO GR THAN MAXTIME | 50 |
| 51 |                             |       | 12D                                    | 51 |
| 52 | USEMAXDT                    | DLOAD | POOHCHK                                | 52 |
| 53 |                             |       | SIGN                                   | 53 |
| 54 |                             |       | 12D                                    | 54 |
| 55 |                             |       | DT/2                                   | 55 |
| 56 |                             |       |                                        | 56 |
| 57 |                             |       |                                        | 57 |
| 58 |                             |       |                                        | 58 |
| 59 |                             |       |                                        | 59 |
| 60 |                             |       |                                        | 60 |

|          |                |                             |                                       |
|----------|----------------|-----------------------------|---------------------------------------|
|          | STCALL         | DT/2<br>POOHCHK             |                                       |
| MAXDT    | DLOAD          | PDDL<br>DT/2MAX             | # EXCHANGE DT/2MAX WITH COMPUTED MAX. |
|          | GOTO           |                             |                                       |
| GETMAXDT | RTB            | DT/2COMP                    |                                       |
|          |                | SIGNMPAC                    |                                       |
|          | STCALL         | DT/2<br>USEMAXDT            |                                       |
| POOHCHK  | DLOAD          | ABS                         |                                       |
|          | DSU            | DT/2<br>BMN<br>DT/2MIN      |                                       |
|          | SLOAD          | A-PCHK<br>BHIZ<br>MODREG    |                                       |
|          |                | +3                          |                                       |
|          | GOTO           |                             |                                       |
|          |                | TIMESTEP                    |                                       |
|          | BON            |                             | # WAS THIS CALL VIA CSM(LEM)PREC      |
|          |                | PRECIFLG<br>TIMESTEP        | # YES                                 |
|          | DLOAD          | DSU<br>DT/2<br>12D          |                                       |
|          | BMN            | BOFCLR<br>A-PCHK<br>NEWIFLG |                                       |
|          |                | TIMESTEP                    |                                       |
|          | DLOAD          | DSU<br>TDEC                 |                                       |
|          |                | TET                         |                                       |
|          | BMN            |                             | # NO BACKWARD INTEGRATION             |
|          |                | INTEXIT                     |                                       |
|          | PDDL           | SR4                         |                                       |
|          |                | DT/2                        | # IS 4(DT) LS (TDEC - TET)            |
|          | SR2R           | BDSU                        |                                       |
|          | BMN            | GOTO<br>INTEXIT<br>TIMESTEP |                                       |
| DT/2MIN  | 2DEC           | 3 B-20                      |                                       |
| DT/2MAX  | 2DEC           | 4000 E2 B-20                |                                       |
| INTSTALL | EXIT<br>CA     | RASFLAG                     |                                       |
|          | MASK<br>EXTEND | INTBITAB                    | # IS THIS STALL AREA FREE             |
|          | BZF            | OKTOGRAB                    | # YES                                 |

|    |          |        |             |                                             |    |
|----|----------|--------|-------------|---------------------------------------------|----|
| 1  |          |        |             |                                             | 1  |
| 2  |          | CAF    | WAKESTAL    |                                             | 2  |
| 3  |          | TC     | JOBSLEEP    |                                             | 3  |
| 4  | INTWAKE0 | EXIT   |             |                                             | 4  |
| 5  |          | TCF    | INTWAKE1    |                                             | 5  |
| 6  |          |        |             |                                             | 6  |
| 7  | INTWAKE  | CS     | RASFLAG     | # IS THIS INTSTALLED ROUTINE TO BE          | 7  |
| 8  |          | MASK   | REINTBIT    | # RESTARTED                                 | 8  |
| 9  |          | CCS    | A           |                                             | 9  |
| 10 |          | TC     | INTWAKE1    | # NO                                        | 10 |
| 11 |          |        |             |                                             | 11 |
| 12 |          | INDEX  | FIXLOC      |                                             | 12 |
| 13 |          | CA     | QPRET       |                                             | 13 |
| 14 |          | TS     | TBASE2      | # YES, DON'T RESTART WITH SOMEONE ELSE'S Q  | 14 |
| 15 |          |        |             |                                             | 15 |
| 16 |          | TC     | PHASCHNG    |                                             | 16 |
| 17 |          | OCT    | 04022       |                                             | 17 |
| 18 |          |        |             |                                             | 18 |
| 19 |          | CA     | TBASE2      |                                             | 19 |
| 20 |          | INDEX  | FIXLOC      |                                             | 20 |
| 21 |          | TS     | QPRET       |                                             | 21 |
| 22 |          |        |             |                                             | 22 |
| 23 |          | CAF    | REINTBIT    |                                             | 23 |
| 24 |          | MASK   | RASFLAG     |                                             | 24 |
| 25 |          | EXTEND |             |                                             | 25 |
| 26 |          | BZF    | GOBAC       | # DON'T INTWAKE IF WE CAME HERE VIA RESTART | 26 |
| 27 |          |        |             |                                             | 27 |
| 28 | INTWAKE1 | CAF    | WAKESTAL    |                                             | 28 |
| 29 |          | INHINT |             |                                             | 29 |
| 30 |          | TC     | JOBWAKE     |                                             | 30 |
| 31 |          | CCS    | LOCCTR      |                                             | 31 |
| 32 |          | TCF    | INTWAKE1    |                                             | 32 |
| 33 | FORTYONE | DEC    | 41          |                                             | 33 |
| 34 |          | CS     | INTBITAB    |                                             | 34 |
| 35 |          | MASK   | RASFLAG     |                                             | 35 |
| 36 |          | TS     | RASFLAG     | # RELEASE STALL AREA                        | 36 |
| 37 |          | RELINT |             |                                             | 37 |
| 38 |          | TCF    | GOBAC       |                                             | 38 |
| 39 | OKTOGRAB | CAF    | INTFLBIT    |                                             | 39 |
| 40 |          | INHINT |             |                                             | 40 |
| 41 |          | ADS    | RASFLAG     |                                             | 41 |
| 42 | GOBAC    | TC     | INTPRET     |                                             | 42 |
| 43 |          | RVQ    |             |                                             | 43 |
| 44 | WAKESTAL | CADR   | INTSTALL +1 |                                             | 44 |
| 45 | INTBITAB | OCT    | 20100       |                                             | 45 |
| 46 |          |        |             |                                             | 46 |
| 47 |          |        |             |                                             | 47 |
| 48 |          |        |             |                                             | 48 |
| 49 |          |        |             |                                             | 49 |
| 50 |          |        |             |                                             | 50 |
| 51 |          |        |             |                                             | 51 |
| 52 |          |        |             |                                             | 52 |
| 53 |          |        |             |                                             | 53 |
| 54 |          |        |             |                                             | 54 |
| 55 |          |        |             |                                             | 55 |
| 56 |          |        |             |                                             | 56 |
| 57 |          |        |             |                                             | 57 |
| 58 |          |        |             |                                             | 58 |
| 59 |          |        |             |                                             | 59 |
| 60 |          |        |             |                                             | 60 |

# AVETOMID

#

# THIS ROUTINE PERFORMS THE TRANSITION FROM A THRUSTING PHASE TO THE COAST  
# PHASE BY INITIALIZING THIS VEHICLE'S PERMANENT STATE VECTOR WITH THE  
# VALUES LEFT BY THE AVERAGEG ROUTINE IN RN,VN,PIPTIME.

#

# BEFORE THIS IS DONE THE W-MATRIX, IF ITS VALID (OR WFLAG OR RENDWFLT IS  
# SET) IS INTEGRATED FORWARD TO PIPTIME WITH THE PRE-THRUST STATE VECTOR.

#

# IN ADDITION, THE OTHER VEHICLE IS INTEGRATED (PERMANENT) TO PIPTIME.

#

# FINALLY TRKMKCNT IS ZEROED.

SETLOC INTINIT

BANK

COUNT\* \$\$/INTIN

AVETOMID

STQ

BON

EGRESS

RENDWFLG

INT/W

# W-MATRIX VALID, GO INTEGRATE IT

BON

ORBWFLAG

INT/W

# W-MATRIX VALID, GO INTEGRATE IT.

OTHERS

DLOAD

CALL

# GET SET FOR OTHER VEHICLE INTEGRATION

PIPTIME

# DESIRED TIME

SET

INTSTALL

CALL

VINTFLAG

# CM

SETIFLGS

# SETS UP NONE W-MAT. PERMANENT INTEG.

STCALL

TDEC1

INTEGRV

AXT,2

CALL

# NOW MOVE PROPERLY SCALE RN,UN AS WELL AS

2

# PIPTIME TO INTEGRATION ERASABLES.

BON

INTSTALL

AXT,2

MOONTHIS

+2

0

VLOAD

VSR\*

RN

0,2

STORE

RRECT

STODL

RCV

PIPTIME

STOVL

TET

VN

|    |  |
|----|--|
| 1  |  |
| 2  |  |
| 3  |  |
| 4  |  |
| 5  |  |
| 6  |  |
| 7  |  |
| 8  |  |
| 9  |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |
| 21 |  |
| 22 |  |
| 23 |  |
| 24 |  |
| 25 |  |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 |  |
| 31 |  |
| 32 |  |

# MIDTOAV1  
#

# THIS ROUTINE INTEGRATES (PRECISION) TO THE TIME SPECIFIED IN TDEC1.  
# IF, AT THE END OF AN INTEGRATION TIME STEP, CURRENT TIME PLUS A DELTA  
# TIME (SEE TIMEDELT.....BASED ON THE COMPUTATION TIME FOR ONE TIME STEP)  
# IS GREATER THAN THE DESIRED TIME, ALARM 1703 IS SET AND THE INTEGRATION  
# IS DONE TO THE CURRENT TIME.  
# RETURN IS IN BASIC TO THE RETURN ADDRESS PLUS ONE.

#  
# IF THE INTEGRATION IS FINISHED TO THE DESIRED TIME, RETURN IS IN BASIC  
# TO THE RETURN ADDRESS.

#  
# IN EITHER CASE, BEFORE RETURNING, THE EXTRAPOLATED STATE VECTOR IS TRANSFERRED  
# FROM R,VATT TO R,VN1-PIPTIME1 IS SET TO THE FINISHING INTEGRATION  
# TIME AND MPAC IS SET TO THE DELTA TIME --  
# TAT MINUS CURRENT TIME

# MIDTOAV2  
#

# THIS ROUTINE INTEGRATES THIS VEHICLE'S STATE VECTOR TO THE CURRENT TIME.  
# NO INPUTS ARE REQUIRED OF THE CALLER. RETURN IS IN BASIC TO THE RETURN  
# ADDRESS WITH THE ABOVE TRANSFERS TO R,VN1-PIPTIME1-AND MPAC DONE

|          |               |                                                      |                                           |
|----------|---------------|------------------------------------------------------|-------------------------------------------|
| MIDTOAV2 | EBANK=<br>STQ | IRETURN1<br>CLRGO<br>IRETURN1<br>MID1FLAG<br>ENTMID2 | # INTEGRATE TO PRESENT TIME PLUS TIMEDELT |
|----------|---------------|------------------------------------------------------|-------------------------------------------|

|          |     |                             |                      |
|----------|-----|-----------------------------|----------------------|
| MIDTOAV1 | STQ | SET<br>IRETURN1<br>MID1FLAG | # INTEGRATE TO TDEC1 |
|----------|-----|-----------------------------|----------------------|

|     |                             |                                         |
|-----|-----------------------------|-----------------------------------------|
| RTB | DAD<br>LOADTIME<br>TIMEDELT | # INITIAL CHECK, IS TDEC1 IN THE FUTURE |
|-----|-----------------------------|-----------------------------------------|

|      |                         |       |
|------|-------------------------|-------|
| BDSU | BPL<br>TDEC1<br>ENTMID1 | # Y5S |
|------|-------------------------|-------|

|      |        |                                     |
|------|--------|-------------------------------------|
| CALL | NOTIME | # NO, SET ALARM, SWITCH TO MIDTOAV2 |
|------|--------|-------------------------------------|

|         |     |                                      |
|---------|-----|--------------------------------------|
| ENTMID2 | RTB | DAD<br>LOADTIME<br>TIMEDELT<br>TDEC1 |
|---------|-----|--------------------------------------|

|         |      |                  |
|---------|------|------------------|
| ENTMID1 | CALL | INTSTALL<br>CALL |
|---------|------|------------------|



|    |  |       |       |    |
|----|--|-------|-------|----|
| 1  |  |       |       | 1  |
| 2  |  | DIMOF | FLAG  | 2  |
| 3  |  | THIS  | VINT  | 3  |
| 4  |  |       |       | 4  |
| 5  |  | CLEAR | SET   | 5  |
| 6  |  |       | INTYP | 6  |
| 7  |  |       | FLG   | 7  |
| 8  |  |       | MIDAV | 8  |
| 9  |  |       | FLG   | 9  |
| 10 |  |       |       | 10 |
| 11 |  |       |       | 11 |
| 12 |  |       |       | 12 |
| 13 |  |       |       | 13 |
| 14 |  |       |       | 14 |
| 15 |  |       |       | 15 |
| 16 |  |       |       | 16 |
| 17 |  |       |       | 17 |
| 18 |  |       |       | 18 |
| 19 |  |       |       | 19 |
| 20 |  |       |       | 20 |
| 21 |  |       |       | 21 |
| 22 |  |       |       | 22 |
| 23 |  |       |       | 23 |
| 24 |  |       |       | 24 |
| 25 |  |       |       | 25 |
| 26 |  |       |       | 26 |
| 27 |  |       |       | 27 |
| 28 |  |       |       | 28 |
| 29 |  |       |       | 29 |
| 30 |  |       |       | 30 |
| 31 |  |       |       | 31 |
| 32 |  |       |       | 32 |
| 33 |  |       |       | 33 |
| 34 |  |       |       | 34 |
| 35 |  |       |       | 35 |
| 36 |  |       |       | 36 |
| 37 |  |       |       | 37 |
| 38 |  |       |       | 38 |
| 39 |  |       |       | 39 |
| 40 |  |       |       | 40 |
| 41 |  |       |       | 41 |
| 42 |  |       |       | 42 |
| 43 |  |       |       | 43 |
| 44 |  |       |       | 44 |
| 45 |  |       |       | 45 |
| 46 |  |       |       | 46 |
| 47 |  |       |       | 47 |
| 48 |  |       |       | 48 |
| 49 |  |       |       | 49 |
| 50 |  |       |       | 50 |
| 51 |  |       |       | 51 |
| 52 |  |       |       | 52 |
| 53 |  |       |       | 53 |
| 54 |  |       |       | 54 |
| 55 |  |       |       | 55 |
| 56 |  |       |       | 56 |
| 57 |  |       |       | 57 |
| 58 |  |       |       | 58 |
| 59 |  |       |       | 59 |
| 60 |  |       |       | 60 |

```
1
2 BMN GOTO
3 A-PCHK
4 TIMEINC
5
6 NOTIME CLEAR EXIT # TOO LATE
7
8 INCR MID1FLAG
9 TC IRETURN1 # SET ERROR EXIT (CALLOC +2)
10 ALARM # INSUFFICIENT TIME FOR INTEGRATION --
11 OCT 1703 #
12 TC INTPRET # TIG WILL BE SLIPPED...
13 RVQ
14
15 3CSECS 2DEC 3
16
17 TIMEDELT 2DEC 2000
18
19 BANK 27
20 SETLOC UPDATE2
21 BANK
22 EBANK= INTWAKUQ
23
24 COUNT* $$/INTIN
25
26 INTWAKUQ = INTWAK1Q # TEMPORARY UNTIL NAME OF INTWAK1Q IS CHNG
27
28 INTWAKEU RELINT
29 EXTEND
30 QXCH INTWAKUQ # SAVE Q FOR RETURN
31
32 TC INTPRET
33
34 SLOAD BZE # IS THIS A CSM/LEM STATE VECTOR UPDATE
35 UPSVFLAG # REQUEST. IF NOT GO TO INTWAKUP.
36 INTWAKUP
37
38 VLOAD # MOVE PRECT(6) AND VRECT(6) INTO
39 # RCV(6) AND VCV(6) RESPECTIVELY.
40 RRECT
41 STOVL RCV
42 VRECT
43 # NOW GO TO 'RECTIFY +13D' TO
44 # STORE VRECT INTO VCV AND ZERO OUT
45 # TDELTA(6),TNUV(6),TC(2), AND XKEP(2)
46 SLOAD ABS
47 UPSVFLAG # COMPARE ABSOLUTE VALUE OF 'UPSVFLAG'
48 DSU # TO 'UPDATE MOON STATE VECTOR CODE'
49 BZE # TO DETERMINE WHETHER THE STATE VECTOR TO
50 UPMNSVCD # BE UPDATED IS IN THE EARTH OR LUNAR
51 INTWAKEM # SPHERE OF INFLUENCE.....
52 AXT,2 # EARTH SPHERE OF INFLUENCE.
53 DEC 0
54 MOONFLAG
55
56
57
58
59
60
```

|    |                                      |       |          |                                          |    |  |  |
|----|--------------------------------------|-------|----------|------------------------------------------|----|--|--|
| 1  |                                      |       |          |                                          | 1  |  |  |
| 2  |                                      |       |          |                                          | 2  |  |  |
| 3  | INTWAKEM                             | AXT,2 | INTWAKEC | # LUNAR SPHERE OF INFLUENCE.             | 3  |  |  |
| 4  |                                      | DEC   | SET      |                                          | 4  |  |  |
| 5  | INTWAKEC                             | SLOAD | 2        | # COMMON CODING AFTER X2 INITIALIZED AND | 5  |  |  |
| 6  |                                      |       | MOONFLAG |                                          | 6  |  |  |
| 7  | # MOONFLAG SET (OR CLEARED).         |       |          |                                          | 7  |  |  |
| 8  | # IS THIS A REQUEST FOR A LEM OR CSM |       |          |                                          | 8  |  |  |
| 9  | # STATE VECTOR UPDATE.....           |       |          |                                          | 9  |  |  |
| 10 |                                      | CALL  | INTWAKLM | # UPDATE CSM STATE VECTOR                | 10 |  |  |
| 11 |                                      |       | ATOPCSM  |                                          | 11 |  |  |
| 12 |                                      |       |          |                                          | 12 |  |  |
| 13 |                                      | CLEAR | GOTO     |                                          | 13 |  |  |
| 14 |                                      |       | ORBWFLAG |                                          | 14 |  |  |
| 15 |                                      |       |          |                                          | 15 |  |  |
| 16 | INTWAKLM                             | CALL  | INTWAKEX | # UPDATE LM STATE VECTOR                 | 16 |  |  |
| 17 |                                      |       | ATOPLEM  |                                          | 17 |  |  |
| 18 |                                      |       |          |                                          | 18 |  |  |
| 19 | INTWAKEX                             | CLEAR |          |                                          | 19 |  |  |
| 20 |                                      |       | RENDWFLG |                                          | 20 |  |  |
| 21 |                                      |       |          |                                          | 21 |  |  |
| 22 | INTWAKUP                             | SSP   | CALL     | # REMOVE `UPDATE STATE VECTOR INDICATOR' | 22 |  |  |
| 23 |                                      |       | UPSVFLAG |                                          | 23 |  |  |
| 24 |                                      |       |          |                                          | 24 |  |  |
| 25 |                                      |       | 0        | # RELEASE `GRAB' OF ORBIT INTEG.         | 25 |  |  |
| 26 |                                      |       | INTWAKEO |                                          | 26 |  |  |
| 27 | EXIT                                 |       |          |                                          | 27 |  |  |
| 28 |                                      |       |          |                                          | 28 |  |  |
| 29 |                                      | TC    | PHASCHNG |                                          | 29 |  |  |
| 30 |                                      | OCT   | 04026    |                                          | 30 |  |  |
| 31 |                                      | TC    | INTWAKUQ |                                          | 31 |  |  |
| 32 |                                      |       |          |                                          | 32 |  |  |
| 33 | UPMNSVCD                             | OCT   | 2        |                                          | 33 |  |  |
| 34 |                                      | OCT   | 0        |                                          | 34 |  |  |
| 35 |                                      |       |          |                                          | 35 |  |  |
| 36 | GRP2PC                               | STQ   | EXIT     |                                          | 36 |  |  |
| 37 |                                      |       | GRP2SVQ  |                                          | 37 |  |  |
| 38 |                                      | TC    | PHASCHNG |                                          | 38 |  |  |
| 39 |                                      | OCT   | 04022    |                                          | 39 |  |  |
| 40 |                                      | TC    | INTPRET  |                                          | 40 |  |  |
| 41 |                                      | GOTO  |          |                                          | 41 |  |  |
| 42 | GRP2SVQ                              |       |          |                                          | 42 |  |  |
| 43 |                                      |       |          |                                          | 43 |  |  |
| 44 |                                      |       |          |                                          | 44 |  |  |
| 45 |                                      |       |          |                                          | 45 |  |  |
| 46 |                                      |       |          |                                          | 46 |  |  |
| 47 |                                      |       |          |                                          | 47 |  |  |
| 48 |                                      |       |          |                                          | 48 |  |  |
| 49 |                                      |       |          |                                          | 49 |  |  |
| 50 |                                      |       |          |                                          | 50 |  |  |
| 51 |                                      |       |          |                                          | 51 |  |  |
| 52 |                                      |       |          |                                          | 52 |  |  |
| 53 |                                      |       |          |                                          | 53 |  |  |
| 54 |                                      |       |          |                                          | 54 |  |  |
| 55 |                                      |       |          |                                          | 55 |  |  |
| 56 |                                      |       |          |                                          | 56 |  |  |
| 57 |                                      |       |          |                                          | 57 |  |  |
| 58 |                                      |       |          |                                          | 58 |  |  |
| 59 |                                      |       |          |                                          | 59 |  |  |
| 60 |                                      |       |          |                                          | 60 |  |  |

# ORBITAL\_INTEGRATION

# ORBITAL INTEGRATION

# DELETE

BANK 13  
SETLOC ORBITAL  
BANK  
COUNT\* \$\$/ORBIT

# DELETE  
KEPPREP

LXA,2 SETPD  
PBODY

DLOAD\* 0  
MUEARTH,2

# Sqrt(MU) (+18 OR +15) 0D PL 2D

PDVL UNIT

# PL 8D

PDDL RCV

# NORM R (+29 OR +27 - N1) 2D PL 4D

36D  
X1

PDVL

DOT

PDDL

# F\*SQRT(MU) (+7 OR +5) 4D PL 6D

VCV

TAU.

# (+28)

DSU

NORM

TC

S1

SR1

DDV

PDDL

2D

DMP

PUSH

# FS (+6 +N1-N2) 6D PL 8D

4D

DSQ

PDDL

# (FS)SQ (+12 +2(N1-N2)) 8D PL 10D

4D

DSQ

PDDL\*

# SSQ/MU (-20R +2(N1-N2)) 10D PL 12D

MUEARTH,2

SR3

PDVL

SR4

VSQ

# PREALIGN MU (+43 OR +37) 12D PL 14D

VCV

DMP

BDSU

# PL 12D

DDV

36D

DMP

DMP

# PL 10D

DMP

2D

# -(1/R-ALPHA) (+12 +3N1-2N2)

SL\*

DP2/3

XSU,1

0

-3,1

# 10L(1/R-ALPHA) (+13 +2(N1-N2))

DAD

# 2(FS)SQ - ETCETERA PL 8D

S1

# X1 = N2-N1

SL\*

DSU

# -FS+2(FS)SQ ETC (+6 +N1-N2) PL 6D

8D,1

DMP

DMP

0D

SL\*

4D

SL\*

1412THE



|    |      |        |          |    |
|----|------|--------|----------|----|
| 1  |      |        |          | 1  |
| 2  | FBR3 | LXA,1  | SSP      | 2  |
| 3  |      |        | DIFEQCNT | 3  |
| 4  |      |        | S1       | 4  |
| 5  |      | DEC    | -13      | 5  |
| 6  |      | DLOAD  | SR       | 6  |
| 7  |      |        | DT/2     | 7  |
| 8  |      |        | 9D       | 8  |
| 9  |      | TIX,1  | ROUND    | 9  |
| 10 |      |        | +1       | 10 |
| 11 |      | PUSH   | DAD      | 11 |
| 12 |      |        | TC       | 12 |
| 13 |      | STODL  | TAU.     | 13 |
| 14 |      | DAD    |          | 14 |
| 15 |      |        | TET      | 15 |
| 16 |      | STCALL | TET      | 16 |
| 17 |      |        | KEPPREP  | 17 |
| 18 |      |        |          | 18 |
| 19 |      |        |          | 19 |
| 20 |      |        |          | 20 |
| 21 |      |        |          | 21 |
| 22 |      |        |          | 22 |
| 23 |      |        |          | 23 |
| 24 |      |        |          | 24 |
| 25 |      |        |          | 25 |
| 26 |      |        |          | 26 |
| 27 |      |        |          | 27 |
| 28 |      |        |          | 28 |
| 29 |      |        |          | 29 |
| 30 |      |        |          | 30 |
| 31 |      |        |          | 31 |
| 32 |      |        |          | 32 |
| 33 |      |        |          | 33 |
| 34 |      |        |          | 34 |
| 35 |      |        |          | 35 |
| 36 |      |        |          | 36 |
| 37 |      |        |          | 37 |
| 38 |      |        |          | 38 |
| 39 |      |        |          | 39 |
| 40 |      |        |          | 40 |
| 41 |      |        |          | 41 |
| 42 |      |        |          | 42 |
| 43 |      |        |          | 43 |
| 44 |      |        |          | 44 |
| 45 |      |        |          | 45 |
| 46 |      |        |          | 46 |
| 47 |      |        |          | 47 |
| 48 |      |        |          | 48 |
| 49 |      |        |          | 49 |
| 50 |      |        |          | 50 |
| 51 |      |        |          | 51 |
| 52 |      |        |          | 52 |
| 53 |      |        |          | 53 |
| 54 |      |        |          | 54 |
| 55 |      |        |          | 55 |
| 56 |      |        |          | 56 |
| 57 |      |        |          | 57 |
| 58 |      |        |          | 58 |
| 59 |      |        |          | 59 |
| 60 |      |        |          | 60 |

# ORBITAL\_INTEGRATION

# AGC ROUTINE TO COMPUTE ACCELERATION COMPONENTS.

|        |                 |                                 |
|--------|-----------------|---------------------------------|
| ACCOMP | LXA,1           | LXA,2<br>PBODY<br>PBODY         |
|        | VLOAD           |                                 |
|        | STOVL           | ZEROVEC<br>FV                   |
|        | VSL*            | ALPHAV<br>VAD<br>0        -7,2  |
|        | STORE<br>BOF    | RCV<br>BETAV<br>XCHX,2          |
|        |                 | DIMOF<br>FLAG<br>+5<br>DIFEQCNT |
|        | STORE<br>XCHX,2 | VECTAB,2                        |
|        |                 | DIFEQCNT                        |
|        | VLOAD           | UNIT<br>ALPHAV                  |
|        | STODL           | ALPHAV                          |
|        | STORE<br>CALL   | 36D<br>ALPHAM                   |
|        | VLOAD           | GAMCOMP<br>SXA,1<br>BETAV       |
|        | STODL           | S2<br>ALPHAV<br>BETAM           |
|        | STORE<br>BOF    | ALPHAM<br>DLOAD<br>MIDFLAG      |
|        | CALL            | OBLATE<br>TET                   |
|        | AXT,2           | LSPOS<br>LXA,1<br>2             |
|        | BOF             | S2                              |
|        |                 | MOONFLAG<br>+3                  |
|        | VCOMP           | AXT,2<br>0                      |
|        | STORE<br>STOVL  | BETAV<br>RPQV                   |

|    |         |        |                                    |
|----|---------|--------|------------------------------------|
| 1  |         |        |                                    |
| 2  |         |        | 2D                                 |
| 3  |         | STORE  | RPSV                               |
| 4  |         | SLOAD  | DSU                                |
| 5  |         |        | MODREG                             |
| 6  |         |        | OCT27                              |
| 7  |         | BHIZ   | BOF                                |
| 8  |         |        | +3                                 |
| 9  |         |        | DIM0FLAG                           |
| 10 |         | VLOAD  | GETRPSV                            |
| 11 |         |        | VXSC                               |
| 12 |         |        | ALPHAV                             |
| 13 |         | VSR*   | ALPHAM                             |
| 14 |         |        | VSU                                |
| 15 |         |        | 1,2                                |
| 16 |         |        | BETAV                              |
| 17 |         | XCHX,2 |                                    |
| 18 |         |        | DIFEQCNT                           |
| 19 |         | STORE  | VECTAB +6,2                        |
| 20 |         | STORE  | RQVV                               |
| 21 |         | XCHX,2 |                                    |
| 22 |         |        | DIFEQCNT                           |
| 23 | GETRPSV | VLOAD  | INCR,1                             |
| 24 |         |        | RPQV                               |
| 25 |         |        | 4                                  |
| 26 |         | CLEAR  | BOF                                |
| 27 |         |        | RPQFLAG                            |
| 28 |         |        | MOONFLAG                           |
| 29 |         |        | +5                                 |
| 30 |         | VSR    | VAD                                |
| 31 |         |        | 9D                                 |
| 32 |         |        | RPSV                               |
| 33 |         | STORE  | RPSV                               |
| 34 |         | CALL   |                                    |
| 35 |         |        | GAMCOMP                            |
| 36 |         | AXT,2  | INCR,1                             |
| 37 |         |        | 4                                  |
| 38 |         |        | 4                                  |
| 39 |         | VLOAD  |                                    |
| 40 |         |        | RPSV                               |
| 41 |         | STCALL | BETAV                              |
| 42 |         |        | GAMCOMP                            |
| 43 |         | GOTO   |                                    |
| 44 |         |        | OBLATE                             |
| 45 | GAMCOMP | VLOAD  | VSR1                               |
| 46 |         |        | BETAV                              |
| 47 |         | VSQ    | SETPD                              |
| 48 |         |        | 0                                  |
| 49 |         | NORM   | ROUND                              |
| 50 |         |        | 31D                                |
| 51 |         | PDDL   | NORM # NORMED B SQUARED TO PD LIST |



|        |          |                                        |
|--------|----------|----------------------------------------|
|        | ALPHAM   | # NORMALIZE (LESS ONE) LENGTH OF ALPHA |
|        | 32D      | # SAVING NORM SCALE FACTOR IN X1       |
| SR1    | PDVL     |                                        |
|        | BETAV    | # C(PDL+2) = ALMOST NORMED ALPHA       |
| UNIT   |          |                                        |
| STODL  | BETAV    |                                        |
|        | 36D      |                                        |
| STORE  | BETAM    |                                        |
| NORM   | BDDV     | # FORM NORMALIZE QUOTIEN ALPHAM/BETAM  |
|        | 33D      |                                        |
| SR1R   | PUSH     | # C(PDL+2) = ALMOST NORMALIZE RHO.     |
| DLOAD* |          |                                        |
|        | ASCALE,1 |                                        |
| STORE  | S1       |                                        |
| XCHX,2 | XAD,2    |                                        |
|        | S1       |                                        |
|        | 32D      |                                        |
| XSU,2  | DLOAD    |                                        |
|        | 33D      |                                        |
|        | 2D       |                                        |
| SR*    | XCHX,2   |                                        |
|        | 0        | -1,2                                   |
|        | S1       |                                        |
| PUSH   | SR1R     | # RHO/4 TO 4D                          |
| PDVL   | DOT      |                                        |
|        | ALPHAV   |                                        |
|        | BETAV    |                                        |
| SL1R   | BDSU     | # (RHO/4) - 2(ALPHAV/2.BETAV/2)        |
| PUSH   | DMPR     | # TO PDL+6                             |
|        | 4        |                                        |
| SL1    |          |                                        |
| PUSH   | DAD      |                                        |
|        | DQUARTER |                                        |
| PUSH   | SQRT     |                                        |
| DMPR   | PUSH     |                                        |
|        | 10D      |                                        |
| SL1    | DAD      |                                        |
|        | DQUARTER |                                        |
| PDDL   | DAD      | # (1/4)+2((Q+1)/4) TO PD+14D           |
|        | 10D      |                                        |
|        | HALFDP   |                                        |
| DMPR   | SL1      |                                        |
|        | 8D       |                                        |
| DAD    | DDV      |                                        |
|        | THREE/8  |                                        |
|        | 14D      |                                        |
| DMPR   | VXSC     |                                        |
|        | 6        |                                        |
|        | BETAV    | #                                      |
| PDVL   | VSR3     | # (G/2)(C(PD+4))B̄/2 TO PD+16D         |

|    |                     |                             |    |
|----|---------------------|-----------------------------|----|
| 1  | # CARDTYPE=INTERCOM |                             | 1  |
| 2  |                     | ALPHAV                      | 2  |
| 3  |                     | PUSH                        | 3  |
| 4  |                     | # A12 + C(PD+16D) TO PD+16D | 4  |
| 5  |                     | DLOAD                       | 5  |
| 6  |                     | 0                           | 6  |
| 7  |                     | 12D                         | 7  |
| 8  |                     | ROUND                       | 8  |
| 9  |                     | 30D                         | 9  |
| 10 |                     | DMP*                        | 10 |
| 11 |                     | 2                           | 11 |
| 12 |                     | MUEARTH,2                   | 12 |
| 13 |                     | VXSC                        | 13 |
| 14 |                     | XCHX,2                      | 14 |
| 15 |                     | XAD,2                       | 15 |
| 16 |                     | S1                          | 16 |
| 17 |                     | S2                          | 17 |
| 18 |                     | XSU,2                       | 18 |
| 19 |                     | XSU,2                       | 19 |
| 20 |                     | 30D                         | 20 |
| 21 |                     | 31D                         | 21 |
| 22 |                     | BOV                         | 22 |
| 23 |                     | # CLEAR OVIND               | 23 |
| 24 |                     | +1                          | 24 |
| 25 |                     | VSR*                        | 25 |
| 26 |                     | XCHX,2                      | 26 |
| 27 |                     | 0                           | 27 |
| 28 |                     | -1,2                        | 28 |
| 29 |                     | S1                          | 29 |
| 30 |                     | VAD                         | 30 |
| 31 |                     | FV                          | 31 |
| 32 |                     | FV                          | 32 |
| 33 |                     | RVQ                         | 33 |
| 34 |                     | # RETURN IF NO OVERFLOW     | 34 |
| 35 |                     | +1                          | 35 |
| 36 | GOBAQUE             | VLOAD                       | 36 |
| 37 |                     | ABVAL                       | 37 |
| 38 |                     | TDELTAV                     | 38 |
| 39 |                     | BZE                         | 39 |
| 40 |                     | INT-ABRT                    | 40 |
| 41 |                     | DLOAD                       | 41 |
| 42 |                     | SR                          | 42 |
| 43 |                     | H                           | 43 |
| 44 |                     | 9D                          | 44 |
| 45 |                     | BDSU                        | 45 |
| 46 |                     | TC                          | 46 |
| 47 |                     | TAU.                        | 47 |
| 48 |                     | TET                         | 48 |
| 49 |                     | DSU                         | 49 |
| 50 |                     | STCALL                      | 50 |
| 51 |                     | TET                         | 51 |
| 52 |                     | KEPPREP                     | 52 |
| 53 |                     | CALL                        | 53 |
| 54 |                     | RECTIFY                     | 54 |
| 55 |                     | SETGO                       | 55 |
| 56 |                     | RPQFLAG                     | 56 |
| 57 |                     | TESTLOOP                    | 57 |
| 58 |                     |                             | 58 |
| 59 | INT-ABRT            | EXIT                        | 59 |
| 60 |                     | TC                          | 60 |
| 61 |                     | P00D00                      | 61 |
| 62 |                     | OCT                         | 62 |
| 63 |                     | 00430                       | 63 |
| 64 |                     |                             | 64 |
| 65 |                     |                             | 65 |
| 66 |                     |                             | 66 |
| 67 |                     |                             | 67 |
| 68 |                     |                             | 68 |
| 69 |                     |                             | 69 |
| 70 |                     |                             | 70 |
| 71 |                     |                             | 71 |
| 72 |                     |                             | 72 |
| 73 |                     |                             | 73 |
| 74 |                     |                             | 74 |
| 75 |                     |                             | 75 |
| 76 |                     |                             | 76 |
| 77 |                     |                             | 77 |
| 78 |                     |                             | 78 |
| 79 |                     |                             | 79 |
| 80 |                     |                             | 80 |

# THE OBLATE ROUTINE COMPUTES THE ACCELERATION DUE TO OBLATENESS. IT USES THE UNIT OF THE VEHICLE  
# POSITION VECTOR FOUND IN ALPHAV AND THE DISTANCE TO THE CENTER IN ALPHAM. THIS IS ADDED TO THE SUM OF THE  
# DISTURBING ACCELERATIONS IN FV AND THE PROPER DIFEQ STAGE IS CALLED VIA X1.

|         |       |          |                           |
|---------|-------|----------|---------------------------|
| OBLATE  | LXA,2 | DLOAD    |                           |
|         |       | PBODY    |                           |
|         |       | ALPHAM   |                           |
|         | SETPD | DSU*     |                           |
|         |       | 0        |                           |
|         | BPL   | RDE,2    | # GET URPV                |
|         |       | BOF      |                           |
|         |       | NBRANCH  |                           |
|         |       | MOONFLAG |                           |
|         |       | COSPHIE  |                           |
|         | VLOAD | PDDL     |                           |
|         |       | ALPHAV   |                           |
|         |       | TET      |                           |
|         | PDDL  | CALL     |                           |
|         |       | 3/5      |                           |
|         |       | R-TO-RP  |                           |
|         | STORE | URPV     |                           |
|         | VLOAD | VXV      |                           |
|         |       | 504LM    |                           |
|         | VAD   | ZUNIT    |                           |
|         |       | VXM      |                           |
|         |       | ZUNIT    |                           |
|         |       | MMATRIX  |                           |
|         | UNIT  |          | # POSSIBLY UNNECESSARY    |
| COMTERM | STORE | UZ       |                           |
|         | DLOAD | DMPR     |                           |
|         |       | COSPHI/2 |                           |
|         |       | 3/32     |                           |
|         | PDDL  | DSQ      | # P2/64 TO PD0            |
|         |       | COSPHI/2 |                           |
|         | DMPR  | DSU      |                           |
|         |       | 15/16    |                           |
|         |       | 3/64     |                           |
|         | PUSH  | DMPR     | # P3/32 TO PD2            |
|         |       | COSPHI/2 |                           |
|         | DMP   | SL1R     |                           |
|         |       | 7/12     |                           |
|         | PDDL  | DMPR     |                           |
|         |       | 0        |                           |
|         |       | 2/3      |                           |
|         | BDSU  | PUSH     | # P4/128 TO PD4           |
|         | DMPR  | DMPR     |                           |
|         |       | COSPHI/2 | # BEGIN COMPUTING P5/1024 |
|         |       | 9/16     |                           |
|         | PDDL  | DMPR     |                           |
|         |       | 2        |                           |
|         |       | 5/128    |                           |

1

|    |       |          |        |                                          |
|----|-------|----------|--------|------------------------------------------|
| 1  |       |          |        | # STORE COSPHI**2 SCALED B2 IN 8D        |
| 2  | PUSH  | DMP      |        | # 5 SCALED B3                            |
| 3  |       | 5/8      |        |                                          |
| 4  | PDDL  | SR2      |        | # PUT 5 COSPHI**2, D5, IN 8D. GET        |
| 5  |       |          |        | # COSPHI**2 D2 FROM 8D                   |
| 6  | DAD   | BDSU     |        | # END UP WITH (1-7 COSPHI**2), B5        |
| 7  |       | 8D       |        | # ADDING COSPHI**2 B4 SAME AS COSPHI**2  |
| 8  |       |          |        | # X 2 D5                                 |
| 9  |       | D1/32    |        | # 1 SCALED B5                            |
| 10 | DMP   | DMP      |        |                                          |
| 11 |       | URPV     |        | # X COMPONENT                            |
| 12 |       | 5/8      |        | # 5 SCALED B3                            |
| 13 | VXSC  | VSL5     |        | # AFTER SHIFT, SCALED B5                 |
| 14 |       | URPV     |        | # VECTOR, B1.                            |
| 15 | PDDL  |          |        | # VECTOR INTO 8D, 10D, 12D, SCALED B5.   |
| 16 |       |          |        | # GET 5 COSPHI**2 OUT OF 8D              |
| 17 | DSU   | DAD      |        |                                          |
| 18 |       | D1/32    |        | # 1 B5                                   |
| 19 |       | 8D       |        | # X COMPONENT (SAME AS MULTIPLYING       |
| 20 |       |          |        | # BY UNITX)                              |
| 21 | STODL | 8D       |        |                                          |
| 22 |       | URPV     |        | # X COMPONENT                            |
| 23 | DMP   | DMP      |        |                                          |
| 24 |       | URPV     | +4     | # Z COMPONENT                            |
| 25 |       | 5/8      |        | # 5 B3 ANSWER B5                         |
| 26 | SL1   | DAD      |        | # FROM 12D FOR Z COMPONENT (SL1 GIVES 10 |
| 27 |       |          |        | # INSTEAD OF 5 FOR COEFFICIENT)          |
| 28 | PDDL  | NORM     |        | # BACK INTO 12D FOR Z COMPONENT.         |
| 29 |       | ALPHAM   |        | # SCALED B27 FOR MOON                    |
| 30 |       | X2       |        |                                          |
| 31 | PUSH  | SLOAD    |        | # STORE IN 14D, DESTROYING URPV          |
| 32 |       |          |        | # X COMPONENT                            |
| 33 |       | E32C31RM |        |                                          |
| 34 | DDV   | VXSC     |        | # IF X2 = 0, DIVISION GIVES B53, VXSC    |
| 35 |       |          |        | # OUT OF 8D B5 GIVES B58                 |
| 36 | VSL*  | VAD      |        | # SHIFT MAKES B61, FOR ADDITION OF       |
| 37 |       |          |        | # VECTOR IN 2D                           |
| 38 |       | 0        | -3,2   |                                          |
| 39 | VSL*  | V/SC     |        | # OPERAND FROM 0D. B108 FOR X1 = 0       |
| 40 |       | 0        | -27D,1 | # FOR X1 = 0, MAKES B88, GIVING B-20     |
| 41 |       |          |        | # FOR RESULT.                            |
| 42 | PDDL  | PDDL     |        |                                          |
| 43 |       | TET      |        |                                          |
| 44 |       | 5/8      |        | # ANY NON-ZERO CONSTANT                  |
| 45 | LXA,2 | CALL     |        | # POSITION IN 0D, TIME IN 6D. X2 LEFT    |
| 46 |       |          |        | # ALONE.                                 |
| 47 |       | PBODY    |        |                                          |
| 48 |       | RP-TO-R  |        |                                          |
| 49 | VAD   | BOV      |        | # OVERFLOW INDICATOR RESET IN "RP-TO-R"  |
| 50 |       | FV       |        |                                          |
| 51 |       | GOBAQUE  |        |                                          |

|    |                        |       |                                           |    |
|----|------------------------|-------|-------------------------------------------|----|
| 1  | # CARDINAL_INFORMATION |       |                                           | 1  |
| 2  |                        | STORE | FV                                        | 2  |
| 3  | NBRANCH                | SLOAD | LXA,1                                     | 3  |
| 4  |                        |       | DIFEQCNT                                  | 4  |
| 5  |                        |       | MPAC                                      | 5  |
| 6  |                        | DMP   | CGOTO                                     | 6  |
| 7  |                        |       | -1/12                                     | 7  |
| 8  |                        |       | MPAC                                      | 8  |
| 9  |                        |       | DIFEQTAB                                  | 9  |
| 10 | COSPHIE                | DLOAD |                                           | 10 |
| 11 |                        |       | ALPHAV +4                                 | 11 |
| 12 |                        | STOVL | COSPHI/2                                  | 12 |
| 13 |                        |       | ZUNIT                                     | 13 |
| 14 |                        | GOTO  |                                           | 14 |
| 15 |                        |       | COMTERM                                   | 15 |
| 16 | DIFEQTAB               | CADR  | DIFEQ+0                                   | 16 |
| 17 |                        | CADR  | DIFEQ+1                                   | 17 |
| 18 |                        | CADR  | DIFEQ+2                                   | 18 |
| 19 | TIMESTEP               | BOF   | VLOAD                                     | 19 |
| 20 |                        |       | MIDFLAG                                   | 20 |
| 21 |                        |       | RECTEST                                   | 21 |
| 22 |                        |       | RCV                                       | 22 |
| 23 |                        | DOT   | DMP                                       | 23 |
| 24 |                        |       | VCV                                       | 24 |
| 25 |                        |       | DT/2 # (R.V) X (DELTA T)                  | 25 |
| 26 |                        | BMN   |                                           | 26 |
| 27 |                        |       | RECTEST                                   | 27 |
| 28 |                        | BON   | BOF                                       | 28 |
| 29 |                        |       | MOONFLAG                                  | 29 |
| 30 |                        |       | LUNSPH                                    | 30 |
| 31 |                        |       | RPQFLAG                                   | 31 |
| 32 |                        |       | EARSPH                                    | 32 |
| 33 |                        | DLOAD | CALL                                      | 33 |
| 34 |                        |       | TET                                       | 34 |
| 35 |                        |       | LSPOS # RPQV IN MPAC                      | 35 |
| 36 |                        | STORE | RPQV # RPQV                               | 36 |
| 37 |                        | LXA,2 |                                           | 37 |
| 38 | INLUNCHK               | BVSU  | PBODY                                     | 38 |
| 39 |                        |       | ABVAL                                     | 39 |
| 40 |                        |       | RCV                                       | 40 |
| 41 |                        | DSU   | BMN                                       | 41 |
| 42 |                        |       | RSPHERE                                   | 42 |
| 43 |                        |       | DOSWITCH                                  | 43 |
| 44 | RECTEST                | VLOAD | ABVAL # RECTIFY IF                        | 44 |
| 45 |                        |       | TDELTAV                                   | 45 |
| 46 |                        | BOV   |                                           | 46 |
| 47 |                        |       | CALLRECT                                  | 47 |
| 48 |                        | DSU   | BPL # 1) EITHER TDELTAV OR TNUV EQUALS OR | 48 |
| 49 |                        |       | 3/4 # EXCEEDS 3/4 IN MAGNITUDE            | 49 |
| 50 |                        |       | CALLRECT #                                | 50 |
| 51 |                        | DAD   | SL* # OR                                  | 51 |
| 52 |                        |       |                                           | 52 |
| 53 |                        |       |                                           | 53 |
| 54 |                        |       |                                           | 54 |
| 55 |                        |       |                                           | 55 |
| 56 |                        |       |                                           | 56 |
| 57 |                        |       |                                           | 57 |
| 58 |                        |       |                                           | 58 |
| 59 |                        |       |                                           | 59 |
| 60 |                        |       |                                           | 60 |

|    |          |          |                                        |    |
|----|----------|----------|----------------------------------------|----|
| 1  |          |          |                                        | 1  |
| 2  |          | 3/4      | #                                      | 2  |
| 3  |          | 0 -7,2   | #                                      | 3  |
| 4  |          |          | #                                      | 4  |
| 5  | DDV      | DSU      | 2) ABVAL(TDELTAV) EQUALS OR EXCEEDS    | 5  |
| 6  |          | 10D      | .01(ABVAL(RCV))                        | 6  |
| 7  |          | RECRATIO |                                        | 7  |
| 8  | BPL      | VLOAD    |                                        | 8  |
| 9  |          | CALLRECT |                                        | 9  |
| 10 |          | TNUV     |                                        | 10 |
| 11 | ABVAL    | DSU      |                                        | 11 |
| 12 | BOV      | 3/4      |                                        | 12 |
| 13 |          | CALLRECT |                                        | 13 |
| 14 | BMN      |          |                                        | 14 |
| 15 |          | INTGRATE |                                        | 15 |
| 16 | CALLRECT | CALL     |                                        | 16 |
| 17 |          | RECTIFY  |                                        | 17 |
| 18 | INTGRATE | VLOAD    |                                        | 18 |
| 19 |          | TNUV     |                                        | 19 |
| 20 | STOVL    | ZV       |                                        | 20 |
| 21 |          | TDELTAV  |                                        | 21 |
| 22 | STORE    | YV       |                                        | 22 |
| 23 | CLEAR    |          |                                        | 23 |
| 24 |          | JSWITCH  |                                        | 24 |
| 25 | DIFEQ0   | VLOAD    |                                        | 25 |
| 26 |          | SSP      |                                        | 26 |
| 27 |          | YV       |                                        | 27 |
| 28 |          | DIFEQCNT |                                        | 28 |
| 29 |          | 0        |                                        | 29 |
| 30 | STODL    | ALPHAV   |                                        | 30 |
| 31 |          | DPZERO   |                                        | 31 |
| 32 | STORE    | H        | # START H AT ZERO. GOES 0(DELT/2)DELT. | 32 |
| 33 | BON      | GOTO     |                                        | 33 |
| 34 |          | JSWITCH  |                                        | 34 |
| 35 |          | DOW..    |                                        | 35 |
| 36 | EARSPH   | ACCOMP   |                                        | 36 |
| 37 |          | GOTO     |                                        | 37 |
| 38 |          | RPQV     |                                        | 38 |
| 39 | LUNSPH   | INLUNCHK |                                        | 39 |
| 40 |          | SR2      |                                        | 40 |
| 41 |          | 10D      |                                        | 41 |
| 42 | DSU      | BMN      |                                        | 42 |
| 43 |          | RSPHERE  |                                        | 43 |
| 44 |          | RECTEST  |                                        | 44 |
| 45 | BOF      | DLOAD    |                                        | 45 |
| 46 |          | RPQFLAG  |                                        | 46 |
| 47 |          | DOSWITCH |                                        | 47 |
| 48 |          | TET      |                                        | 48 |
| 49 | CALL     |          |                                        | 49 |
| 50 |          | LUNPOS   |                                        | 50 |
| 51 | VCOMP    |          |                                        | 51 |
| 52 | STORE    | RPQV     |                                        | 52 |
| 53 |          |          |                                        | 53 |
| 54 |          |          |                                        | 54 |
| 55 |          |          |                                        | 55 |
| 56 |          |          |                                        | 56 |
| 57 |          |          |                                        | 57 |
| 58 |          |          |                                        | 58 |
| 59 |          |          |                                        | 59 |
| 60 |          |          |                                        | 60 |

|    |  |
|----|--|
| 1  |  |
| 2  |  |
| 3  |  |
| 4  |  |
| 5  |  |
| 6  |  |
| 7  |  |
| 8  |  |
| 9  |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |
| 21 |  |
| 22 |  |
| 23 |  |
| 24 |  |
| 25 |  |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 |  |
| 31 |  |
| 32 |  |
| 33 |  |
| 34 |  |
| 35 |  |
| 36 |  |
| 37 |  |
| 38 |  |
| 39 |  |
| 40 |  |
| 41 |  |
| 42 |  |
| 43 |  |
| 44 |  |
| 45 |  |
| 46 |  |
| 47 |  |
| 48 |  |
| 49 |  |
| 50 |  |
| 51 |  |
| 52 |  |
| 53 |  |
| 54 |  |
| 55 |  |
| 56 |  |
| 57 |  |
| 58 |  |
| 59 |  |
| 60 |  |
| 61 |  |
| 62 |  |
| 63 |  |
| 64 |  |
| 65 |  |
| 66 |  |
| 67 |  |
| 68 |  |
| 69 |  |
| 70 |  |
| 71 |  |
| 72 |  |
| 73 |  |
| 74 |  |
| 75 |  |
| 76 |  |
| 77 |  |
| 78 |  |
| 79 |  |
| 80 |  |





|    |                                                                                                                 |       |         |    |
|----|-----------------------------------------------------------------------------------------------------------------|-------|---------|----|
| 1  | # THE RECTIFY SUBROUTINE IS CALLED BY THE INTEGRATION PROGRAM AND OCCASIONALLY BY THE MEASUREMENT INCORPORATION |       |         | 1  |
| 2  | # ROUTINES TO ESTABLISH A NEW CONIC.                                                                            |       |         | 2  |
| 3  |                                                                                                                 |       |         | 3  |
| 4  |                                                                                                                 |       |         | 4  |
| 5  | RECTIFY                                                                                                         | LXA,2 | VLOAD   | 5  |
| 6  |                                                                                                                 |       | PBODY   | 6  |
| 7  |                                                                                                                 |       | TDELTA  | 7  |
| 8  |                                                                                                                 | VSL*  | VAD     | 8  |
| 9  |                                                                                                                 |       | 0 -7,2  | 9  |
| 10 |                                                                                                                 |       | RCV     | 10 |
| 11 |                                                                                                                 | STORE | RRECT   | 11 |
| 12 |                                                                                                                 | STOVL | RCV     | 12 |
| 13 |                                                                                                                 |       | TNUV    | 13 |
| 14 |                                                                                                                 | VSL*  | VAD     | 14 |
| 15 |                                                                                                                 |       | 0 -4,2  | 15 |
| 16 |                                                                                                                 |       | VCV     | 16 |
| 17 | MINIRECT                                                                                                        | STORE | VRECT   | 17 |
| 18 |                                                                                                                 | STOVL | VCV     | 18 |
| 19 |                                                                                                                 |       | ZEROVEC | 19 |
| 20 |                                                                                                                 | STORE | TDELTA  | 20 |
| 21 |                                                                                                                 | STODL | TNUV    | 21 |
| 22 |                                                                                                                 |       | ZEROVEC | 22 |
| 23 |                                                                                                                 | STORE | TC      | 23 |
| 24 |                                                                                                                 | STORE | XKEP    | 24 |
| 25 |                                                                                                                 | RVQ   |         | 25 |
| 26 |                                                                                                                 |       |         | 26 |
| 27 |                                                                                                                 |       |         | 27 |
| 28 |                                                                                                                 |       |         | 28 |
| 29 |                                                                                                                 |       |         | 29 |
| 30 |                                                                                                                 |       |         | 30 |
| 31 |                                                                                                                 |       |         | 31 |
| 32 |                                                                                                                 |       |         | 32 |
| 33 |                                                                                                                 |       |         | 33 |
| 34 |                                                                                                                 |       |         | 34 |
| 35 |                                                                                                                 |       |         | 35 |
| 36 |                                                                                                                 |       |         | 36 |
| 37 |                                                                                                                 |       |         | 37 |
| 38 |                                                                                                                 |       |         | 38 |
| 39 |                                                                                                                 |       |         | 39 |
| 40 |                                                                                                                 |       |         | 40 |
| 41 |                                                                                                                 |       |         | 41 |
| 42 |                                                                                                                 |       |         | 42 |
| 43 |                                                                                                                 |       |         | 43 |
| 44 |                                                                                                                 |       |         | 44 |
| 45 |                                                                                                                 |       |         | 45 |
| 46 |                                                                                                                 |       |         | 46 |
| 47 |                                                                                                                 |       |         | 47 |
| 48 |                                                                                                                 |       |         | 48 |
| 49 |                                                                                                                 |       |         | 49 |
| 50 |                                                                                                                 |       |         | 50 |
| 51 |                                                                                                                 |       |         | 51 |
| 52 |                                                                                                                 |       |         | 52 |
| 53 |                                                                                                                 |       |         | 53 |
| 54 |                                                                                                                 |       |         | 54 |
| 55 |                                                                                                                 |       |         | 55 |
| 56 |                                                                                                                 |       |         | 56 |
| 57 |                                                                                                                 |       |         | 57 |
| 58 |                                                                                                                 |       |         | 58 |
| 59 |                                                                                                                 |       |         | 59 |
| 60 |                                                                                                                 |       |         | 60 |

# THE THREE DIFEQ ROUTINES -- DIFEQ+0, DIFEQ+12, DIFEQ+24 -- ARE ENTERED TO PROCESS THE CONTRIBUTIONS AT THE  
# BEGINNING, MIDDLE, AND END OF THE TIMESTEP, RESPECTIVELY. THE UPDATING IS DONE BY THE NYSTROM METHOD.

DIFEQ+0           VLOAD   VSR3  
                      FV

                  STCALL   PHIV  
DIFEQ+1           VLOAD   DIFEQCOM  
                      VSR1

                      FV  
                  PUSH     VAD  
                      PHIV  
                  STOVL    PSIV  
                  VSR1    VAD  
                      PHIV

                  STCALL   PHIV  
DIFEQ+2           DLOAD   DIFEQCOM  
                      DMPR

                      H  
                      DP2/3  
                  PUSH     VXSC

                      PHIV  
                  VSL1    VAD  
                      ZV

                  VXSC    VAD  
                      H  
                      YV

                  STOVL   YV  
                      FV  
                  VSR3    VAD

                      PSIV  
                  VXSC    VSL1  
                  VAD

                      ZV  
                  STORE    ZV  
                  BOFF    CALL

                      JSWITCH  
                      ENDSTATE  
                      GRP2PC

                  LXA,2   VLOAD  
                          COLREG  
                          ZV

                  VSL3       # ADJUST W-POSITION FOR STORAGE  
                  STORE    W       +54D,2  
                  VLOAD

                      YV  
                  VSL3    BOV  
                      WMATEND

                  STORE   W,2

                  CALL  
                      GRP2PC

142THE

|    |          |        |           |                                 |
|----|----------|--------|-----------|---------------------------------|
| 1  |          |        | ATOPCSM   |                                 |
| 2  |          |        | ATOPLEM   |                                 |
| 3  |          |        |           |                                 |
| 4  | AMOVED   | SET    | SSP       |                                 |
| 5  |          |        | JSWITCH   |                                 |
| 6  |          |        | COLREG    |                                 |
| 7  |          | DEC    | -30       |                                 |
| 8  |          | BOFF   | SSP       |                                 |
| 9  |          |        | D6OR9FLG  |                                 |
| 10 |          |        | NEXTCOL   |                                 |
| 11 |          |        | COLREG    |                                 |
| 12 |          | DEC    | -48       |                                 |
| 13 |          | GOTO   |           |                                 |
| 14 |          |        | NEXTCOL   |                                 |
| 15 |          |        |           |                                 |
| 16 | RELOADSV | DLOAD  |           | # RELOAD TEMPORARY STATE VECTOR |
| 17 |          |        | TDEC      | # FROM PERMANENT IN CASE OF     |
| 18 |          | STCALL | TDEC1     |                                 |
| 19 |          |        | INTEGRV2  | # BY STARTING AT INTEGRV2.      |
| 20 | DIFEQCOM | DLOAD  | DAD       | # INCREMENT H AND DIFEQCNT.     |
| 21 |          |        | DT/2      |                                 |
| 22 |          |        | H         |                                 |
| 23 |          | INCR,1 | SXA,1     |                                 |
| 24 |          | DEC    | -12       |                                 |
| 25 |          |        | DIFEQCNT  | # DIFEQCNT SET FOR NEXT ENTRY.  |
| 26 |          | STORE  | H         |                                 |
| 27 |          | VXSC   | VSR1      |                                 |
| 28 |          |        | FV        |                                 |
| 29 |          | VAD    | VXSC      |                                 |
| 30 |          |        | ZV        |                                 |
| 31 |          |        | H         |                                 |
| 32 |          | VAD    |           |                                 |
| 33 |          |        | YV        |                                 |
| 34 |          | STORE  | ALPHAV    |                                 |
| 35 |          | BON    | GOTO      |                                 |
| 36 |          |        | JSWITCH   |                                 |
| 37 |          |        | DOW..     |                                 |
| 38 |          |        | FBR3      |                                 |
| 39 |          |        |           |                                 |
| 40 | WMATEND  | CLEAR  | CLEAR     |                                 |
| 41 |          |        | DIMOFLLAG | # DON'T INTEGRATE W THIS TIME   |
| 42 |          |        | ORBWFLAG  | # INVALIDATE W                  |
| 43 |          | CLEAR  |           |                                 |
| 44 |          |        | RENDWFLG  |                                 |
| 45 |          | SET    | EXIT      |                                 |
| 46 |          |        | STATEFLG  | # PICK UP STATE VECTOR UPDATE   |
| 47 |          | TC     | ALARM     |                                 |
| 48 |          | OCT    | 421       |                                 |
| 49 |          | TC     | INTPRET   |                                 |
| 50 |          |        |           |                                 |
| 51 |          |        |           |                                 |
| 52 |          |        |           |                                 |
| 53 |          |        |           |                                 |
| 54 |          |        |           |                                 |
| 55 |          |        |           |                                 |
| 56 |          |        |           |                                 |
| 57 |          |        |           |                                 |
| 58 |          |        |           |                                 |
| 59 |          |        |           |                                 |
| 60 |          |        |           |                                 |

GOTO

## TESTLOOP

## # FINISH INTEGRATING STATE VECTOR

```
ORBITAL ROUTINE FOR EXTRAPOLATION OF THE W MATRIX. IT COMPUTES THE SECOND DERIVATIVE OF EACH COLUMN POSITION
VECTOR OF THE MATRIX AND CALLS THE NYSTROM INTEGRATION ROUTINES TO SOLVE THE DIFFERENTIAL EQUATIONS. THE PROGRAM
USES A TABLE OF VEHICLE POSITION VECTORS COMPUTED DURING THE INTEGRATION OF THE VEHICLE'S POSITION AND VELOCITY.
```

```
DOW.. LXA,2 DLOAD*
 PBODY
 MUEARTH,2
 STCALL BETAM
 DOW..1
 STORE FV
 BOF INCR,1
 MIDFLAG
 NBRANCH
 DEC -6
 LXC,2 DLOAD*
 PBODY
 MUEARTH -2,2
 STCALL BETAM
 DOW..1
 BON VSR6
 MOONFLAG
 +1
 VAD
 STCALL FV
 FV
 NBRANCH
DOW..1 VLOAD VSR4
 PDVL* ALPHAV
 UNIT
 VECTAB,1
 PDVL VPROJ
 ALPHAV
 VXSC VSU
 3/4
 PDDL NORM
 36D
 S2
 PUSH DSQ
 DMP
 NORM PDDL
 34D
 BETAM
 DDV
 SR1
 VXSC
 LXA,2 XAD,2
 S2
 S2
 XAD,2 XAD,2
 S2
 34D
 VSL* RVQ
```

0 -8D,2

```


SETITCTR SSP BOFF # SET ITERCTR FOR LAMBERT CALLS. THIS
 # CODING BELONGS IN INITVEL AND IS HERE
 # FOR PURPOSES OF A ONE-MODULE
 # REMANUFACTURE ONLY. CODING SHOULD
 # BE MOVED BACK TO INITVEL FOR LUMINARY 1B
 SSP
 LAMBERT
 GOTO
 ITERCTR
 5
 LAMBERT


```

SETLOC ORBITAL1  
BANK

3/5 2DEC .6 B-2

THREE/8 2DEC .375

.3D 2DEC .3 B-2

3/64 2DEC 3 B-6

DP1/4 2DEC .25

DQUARTER      EQUALS    DP1/4  
POS1/4        EQUALS    DP1/4  
3/32          2DEC      3 B-5

15/16 2DEC 15. B-4

3/4 2DEC 3.0 B-2

7/12 2DEC .5833333333

9/16 2DEC 9 B-4

5/128 2DEC 5 B-7

DPZERO        EQUALS    ZEROVEC  
DP2/3         2DEC      .6666666667

2/3            EQUALS    DP2/3  
OCT27         OCT       27

```
1
2 BANK 13
3 SETLOC ORBITAL2
4 BANK
5 # IT IS VITAL THAT THE FOLLOWING CONSTANTS NOT BE SHUFFLED
6 DEC -11
7 DEC -2
8 DEC -9
9 DEC -6
10 DEC -2
11 DEC -2
12 DEC 0
13 DEC -12
14 DEC -9
15 DEC -4
16 ASCALE DEC -7
17 DEC -6
18 5/8 2DEC 5 B-3
19
20 -1/12 2DEC -.1
21
22 RECRATIO 2DEC .01
23
24 RSPHERE 2DEC 64373.76 E3 B-29
25
26 RDM 2DEC 16093.44 E3 B-27
27
28 RDE 2DEC 80467.20 E3 B-29
29
30 RATT EQUALS 00
31 VATT EQUALS 6D
32 TAT EQUALS 12D
33 RATT1 EQUALS 14D
34 VATT1 EQUALS 20D
35 MU(P) EQUALS 26D
36 TDEC1 EQUALS 32D
37 URPV EQUALS 14D
38 COSPHI/2 EQUALS URPV +4
39 UZ EQUALS 20D
40 TVEC EQUALS 26D
41
42 QUALITY1 BOF DLOAD
43 MOONFLAG
44 NBRANCH
45 URPV
46
47 QUALITY2 DSQ PDDL DSQ +2 # SQUARE INTO 2D, B2
48 URPV +2 # Y COMPONENT, B1
49
50 DSU DMP VXSC # 5(Y**2-X**2)UR
51 5/8 # CONSTANT, 5B3
52 URPV # VECTOR. RESULT MAXIMUM IS 5, SCALING
```





|    |       |          |                                           |  |
|----|-------|----------|-------------------------------------------|--|
| 1  |       |          |                                           |  |
| 2  |       |          | # HERE B6                                 |  |
| 3  | VSL3  | PDDL     | # STORE SCALED B3 IN 2D, 4D, 6D FOR XYZ   |  |
| 4  |       | URPV     | # X COMPONENT, B1                         |  |
| 5  | SR1   | DAD      | # 2 X X COMPONENT FOR B3 SCALING          |  |
| 6  |       | 2D       | # ADD TO VECTOR X COMPONENT OF ANSWER,    |  |
| 7  |       |          | # SAME AS MULTIPLYING BY UNITX. MAX IS 7. |  |
| 8  | STODL | 2D       |                                           |  |
| 9  |       | URPV +2  | # Y COMPONENT, B1                         |  |
| 10 | SR1   | BDSU     | # 2 X Y COMPONENT FOR B3 SCALING          |  |
| 11 |       | 4D       | # SUBTRACT FROM VECTOR Y COMPONENT OF     |  |
| 12 |       |          | # ANSWER, SAME AS MULTIPLYING BY UNITY.   |  |
| 13 |       |          | # MAX IS 7.                               |  |
| 14 | STORE | 4D       | # 2D HAS VECTOR, B3.                      |  |
| 15 | SLOAD | VXSC     | # MULTIPLY COEFFICIENT TIMES VECTOR IN 2D |  |
| 16 |       | E3J22R2M |                                           |  |
| 17 | PDDL  | RVQ      | # J22 TERM X R**4 IN 2D, SCALED B61       |  |
| 18 |       | COSPHI/2 | # SAME AS URPV +4 Z COMPONENT             |  |
| 19 |       |          |                                           |  |
| 20 |       |          |                                           |  |
| 21 |       |          |                                           |  |
| 22 |       |          |                                           |  |
| 23 |       |          |                                           |  |
| 24 |       |          |                                           |  |
| 25 |       |          |                                           |  |
| 26 |       |          |                                           |  |
| 27 |       |          |                                           |  |
| 28 |       |          |                                           |  |
| 29 |       |          |                                           |  |
| 30 |       |          |                                           |  |
| 31 |       |          |                                           |  |
| 32 |       |          |                                           |  |
| 33 |       |          |                                           |  |
| 34 |       |          |                                           |  |
| 35 |       |          |                                           |  |
| 36 |       |          |                                           |  |
| 37 |       |          |                                           |  |
| 38 |       |          |                                           |  |
| 39 |       |          |                                           |  |
| 40 |       |          |                                           |  |
| 41 |       |          |                                           |  |
| 42 |       |          |                                           |  |
| 43 |       |          |                                           |  |
| 44 |       |          |                                           |  |
| 45 |       |          |                                           |  |
| 46 |       |          |                                           |  |
| 47 |       |          |                                           |  |
| 48 |       |          |                                           |  |
| 49 |       |          |                                           |  |
| 50 |       |          |                                           |  |
| 51 |       |          |                                           |  |
| 52 |       |          |                                           |  |
| 53 |       |          |                                           |  |
| 54 |       |          |                                           |  |
| 55 |       |          |                                           |  |
| 56 |       |          |                                           |  |
| 57 |       |          |                                           |  |
| 58 |       |          |                                           |  |
| 59 |       |          |                                           |  |
| 60 |       |          |                                           |  |

```
1 BANK 22
2 SETLOC INFLIGHT
3 BANK
4
5 EBANK= XSM
6
7 # CALCGTA COMPUTES THE GYRO TORQUE ANGLES REQUIRED TO BRING THE STABLE MEMBER INTO THE DESIRED ORIENTATION.
8 #
9 # THE INPUT IS THE DESIRED STABLE MEMBER COORDINATES REFERRED TO PRESENT STABLE MEMBER COORDINATES. THE THREE
10 # HALF-UNIT VECTORS ARE STORED AT XDC, YDC, AND ZDC.
11 #
12 # THE OUTPUTS ARE THE THREE GYRO TORQUE ANGLES TO BE APPLIED TO THE Y, Z, AND X GYROS AND ARE STORED DP AT IGC,
13 # MGC, AND OGC RESPECTIVELY.
14
15
16 CALCGTA COUNT* $$/INFLT
17 ITA DLOAD # PUSHDOWN 00-03, 16D-27D, 34D-37D
18 S2 # XDC = (XD1 XD2 XD3)
19 XDC # YDC = (YD1 YD2 YD3)
20 PDDL PDDL # ZDC = (ZD1 ZD2 ZD3)
21 HI6ZEROS
22 XDC +4
23 DCOMP VDEF
24 UNIT
25 STODL ZPRIME # ZP = UNIT(-XD3 0 XD1) = (ZP1 ZP2 ZP3)
26 ZPRIME
27
28 SR1
29 STODL SINTH # SIN(IGC) = ZP1
30 ZPRIME +4
31
32 SR1
33 STCALL COSTH # COS(IGC) = ZP3
34 ARCTRIG
35
36 STODL IGC # Y GYRO TORQUING ANGLE FRACTION OF REV.
37 XDC +2
38
39 SR1
40 STODL SINTH # SIN(MGC) = XD2
41 ZPRIME
42
43 DMP PDDL
44 XDC +4 # PD00 = (ZP1)(XD3)
45 ZPRIME +4
46
47 DMP DSU
48 XDC # MPAC = (ZP3)(XD1)
49 STADR
50 STCALL COSTH # COS(MGC) = MPAC - PD00
51 ARCTRIG
52
53
54
55
56
57
58
59
60
```

|    |  |        |         |                         |                  |    |
|----|--|--------|---------|-------------------------|------------------|----|
| 1  |  |        |         |                         |                  | 1  |
| 2  |  | STOVL  | MGC     | # Z GYRO TORQUING ANGLE | FRACTION OF REV. | 2  |
| 3  |  |        | ZPRIME  |                         |                  | 3  |
| 4  |  | DOT    |         |                         |                  | 4  |
| 5  |  |        | ZDC     |                         |                  | 5  |
| 6  |  | STOVL  | COSTH   | # COS(OGC) = ZP . ZDC   |                  | 6  |
| 7  |  |        | ZPRIME  |                         |                  | 7  |
| 8  |  | DOT    |         |                         |                  | 8  |
| 9  |  |        | YDC     |                         |                  | 9  |
| 10 |  | STCALL | SINTH   | # SIN(OGC) = ZP . YDC   |                  | 10 |
| 11 |  |        | ARCTRIG |                         |                  | 11 |
| 12 |  |        |         |                         |                  | 12 |
| 13 |  | STCALL | OGC     | # X GYRO TORQUING ANGLE | FRACTION OF REV. | 13 |
| 14 |  |        | S2      |                         |                  | 14 |
| 15 |  |        |         |                         |                  | 15 |
| 16 |  |        |         |                         |                  | 16 |
| 17 |  |        |         |                         |                  | 17 |
| 18 |  |        |         |                         |                  | 18 |
| 19 |  |        |         |                         |                  | 19 |
| 20 |  |        |         |                         |                  | 20 |
| 21 |  |        |         |                         |                  | 21 |
| 22 |  |        |         |                         |                  | 22 |
| 23 |  |        |         |                         |                  | 23 |
| 24 |  |        |         |                         |                  | 24 |
| 25 |  |        |         |                         |                  | 25 |
| 26 |  |        |         |                         |                  | 26 |
| 27 |  |        |         |                         |                  | 27 |
| 28 |  |        |         |                         |                  | 28 |
| 29 |  |        |         |                         |                  | 29 |
| 30 |  |        |         |                         |                  | 30 |
| 31 |  |        |         |                         |                  | 31 |
| 32 |  |        |         |                         |                  | 32 |
| 33 |  |        |         |                         |                  | 33 |
| 34 |  |        |         |                         |                  | 34 |
| 35 |  |        |         |                         |                  | 35 |
| 36 |  |        |         |                         |                  | 36 |
| 37 |  |        |         |                         |                  | 37 |
| 38 |  |        |         |                         |                  | 38 |
| 39 |  |        |         |                         |                  | 39 |
| 40 |  |        |         |                         |                  | 40 |
| 41 |  |        |         |                         |                  | 41 |
| 42 |  |        |         |                         |                  | 42 |
| 43 |  |        |         |                         |                  | 43 |
| 44 |  |        |         |                         |                  | 44 |
| 45 |  |        |         |                         |                  | 45 |
| 46 |  |        |         |                         |                  | 46 |
| 47 |  |        |         |                         |                  | 47 |
| 48 |  |        |         |                         |                  | 48 |
| 49 |  |        |         |                         |                  | 49 |
| 50 |  |        |         |                         |                  | 50 |
| 51 |  |        |         |                         |                  | 51 |
| 52 |  |        |         |                         |                  | 52 |
| 53 |  |        |         |                         |                  | 53 |
| 54 |  |        |         |                         |                  | 54 |
| 55 |  |        |         |                         |                  | 55 |
| 56 |  |        |         |                         |                  | 56 |
| 57 |  |        |         |                         |                  | 57 |
| 58 |  |        |         |                         |                  | 58 |
| 59 |  |        |         |                         |                  | 59 |
| 60 |  |        |         |                         |                  | 60 |

# ARCTRIG COMPUTES AN ANGLE GIVEN THE SINE AND COSINE OF THIS ANGLE.

#

# THE INPUTS ARE SIN/4 AND COS/4 STORED DP AT SINTH AND COSTH.

#

# THE OUTPUT IS THE CALCULATED ANGLE BETWEEN +.5 AND -.5 REVOLUTIONS AND STORED AT THETA. THE OUTPUT IS ALSO

# AVAILABLE AT MPAC.

ARCTRIG            DLOAD    ABS            # PUSHDOWN 16D-21D

DSU            SINTH  
BMN

QTSN45            # ABS(SIN/4) - SIN(45)/4

TRIG1            # IF (-45,45) OR (135,-135)

DLOAD    SL1            # (45,135) OR (-135,-45)

ACOS            COSTH  
SIGN  
SINTH

STORE    THETA            # X = ARCCOS(COS) WITH SIGN(SIN)  
RVQ

TRIG1            DLOAD    SL1            # (-45,45) OR (135,-135)

SINTH

ASIN

STODL    THETA            # X = ARCSIN(SIN) WITH SIGN(SIN)

COSTH

BMN

TRIG2            # IF (135,-135)

DLOAD    RVQ  
THETA

# X = ARCSIN(SIN)    (-45,45)

TRIG2            DLOAD    SIGN            # (135,-135)

HIDPHALF  
SINTH

DSU

STORE    THETA            # X = .5 WITH SIGN(SIN) - ARCSIN(SIN)  
RVQ       THETA            #        (+) - (+) OR (-) - (-)



# INFLIGHT\_ALIGNMENT\_ROUTINES

# SMNB, NBSM, AND AXISROT, WHICH USED TO APPEAR HERE, HAVE BEEN  
# COMBINED IN A ROUTINE CALLED AX\*SR\*T, WHICH APPEARS AMONG THE POWERED  
# FLIGHT SUBROUTINES.

1412THE

```
1 # CALCGA COMPUTES THE CDU DRIVING ANGLES REQUIRED TO BRING THE STABLE MEMBER INTO THE DESIRED ORIENTATION.
2 #
3 # THE INPUTS ARE 1) THE NAVIGATION BASE COORDINATES REFERRED TO ANY COORDINATE SYSTEM. THE THREE HALF-UNIT
4 # VECTORS ARE STORED AT XNB, YNB, AND ZNB. 2) THE DESIRED STABLE MEMBER COORDINATES REFERRED TO THE SAME
5 # COORDINATE SYSTEM ARE STORED AT XSM, YSM, AND ZSM.
6 #
7 # THE OUTPUTS ARE THE THREE CDU DRIVING ANGLES AND ARE STORED SP AT THETAD, THETAD +1, AND THETAD +2.
```

```
10 CALCGA SETPD # PUSHDOWN 00-05, 16D-21D, 34D-37D
```

```
11 0
12 VLOAD VXV
```

```
13 XNB # XNB = OGA (OUTER GIMBAL AXIS)
14 YSM # YSM = IGA (INNER GIMBAL AXIS)
15 UNIT # PDO = UNIT(OGA X IGA) = MGA
```

```
16
17 DOT ITA
18 ZNB
```

```
19 STOVL S2
20 COSTH # COS(OG) = MGA . ZNB
21 0
```

```
22 DOT
```

```
23 STCALL YNB
24 SINTH # SIN(OG) = MGA . YNB
```

```
25 STOVL ARCTRIG
26 OGC
27 0
```

```
28
29 VXV DOT # PROVISION FOR MG ANGLE OF 90 DEGREES
30 XNB
31 YSM
```

```
32 SL1
33 STOVL COSTH # COS(MG) = IGA . (MGA X OGA)
34 YSM
```

```
35 DOT
```

```
36 XNB
37 STCALL SINTH # SIN(MG) = IGA . OGA
38 ARCTRIG
```

```
39 STORE MGC
```

```
40
41 ABS DSU
42 .166...
```

```
43 BPL
44 GIMLOCK1 # IF ANGLE GREATER THAN 60 DEGREES
```

```
46 CALCGA1 VLOAD DOT
47 ZSM
48 0
```

```
49 STOVL COSTH # COS(IG) = ZSM . MGA
50 XSM
```

|    |    |
|----|----|
| 1  | 2  |
| 3  | 4  |
| 5  | 6  |
| 7  | 8  |
| 9  | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |
| 35 | 36 |
| 37 | 38 |
| 39 | 40 |
| 41 | 42 |
| 43 | 44 |
| 45 | 46 |
| 47 | 48 |
| 49 | 50 |
| 51 | 52 |
| 53 | 54 |
| 55 | 56 |
| 57 | 58 |
| 59 | 60 |
| 61 | 62 |
| 63 | 64 |
| 65 | 66 |
| 67 | 68 |
| 69 | 70 |
| 71 | 72 |
| 73 | 74 |
| 75 | 76 |
| 77 | 78 |
| 79 | 80 |

```
1 # AXISGEN COMPUTES THE COORDINATES OF ONE COORDINATE SYSTEM REFERRED TO ANOTHER COORDINATE SYSTEM.
2 #
3 # THE INPUTS ARE 1) THE STAR1 VECTOR REFERRED TO COORDINATE SYSTEM A STORED AT STARAD. 2) THE STAR2 VECTOR
4 # REFERRED TO COORDINATE SYSTEM A STORED AT STARAD +6. 3) THE STAR1 VECTOR REFERRED TO COORDINATE SYSTEM B STORED
5 # AT LOCATION 6 OF THE VAC AREA. 4) THE STAR2 VECTOR REFERRED TO COORDINATE SYSTEM B STORED AT LOCATION 12D OF
6 # THE VAC AREA.
7 #
8 # THE OUTPUT DEFINES COORDINATE SYSTEM A REFERRED TO COORDINATE SYSTEM B. THE THREE HALF-UNIT VECTORS ARE STORED
9 # AT LOCATIONS XDC, XDC +6, XDC +12D, AND STARAD, STARAD +6, STARAD +12D.
```

```
11 AXISGEN AXT,1 SSP # PUSHDOWN 00-30D, 34D-37D
12
13 STARAD +6
14 S1
15 STARAD -6
16
17 SETPD
18 0
19 AXISGEN1 VLOAD* VXV* # 06D UA = S1
20 STARAD +12D,1 # STARAD +00D UB = S1
21 STARAD +18D,1
22 UNIT STARAD +18D,1 # 12D VA = UNIT(S1 X S2)
23 STORE STARAD +06D VB = UNIT(S1 X S2)
24 VLOAD*
25 STARAD +12D,1
26
27 VXV* VSL1
28 STARAD +18D,1 # 18D WA = UA X VA
29 STORE STARAD +24D,1 # STARAD +12D WB = UB X VB
30
31 TIX,1
32 AXISGEN1
33
34 AXC,1 SXA,1
35 6
36 30D
37
38 AXT,1 SSP
39 18D
40 S1
41 6
42
43 AXT,2 SSP
44 6
45 S2
46 2
47
48 AXISGEN2 XCHX,1 VLOAD*
49 30D # X1=-6 X2=+6 X1=-6 X2=+4 X1=-6 X2=+2
50 0,1
```



|       |        |      |             |             |             |
|-------|--------|------|-------------|-------------|-------------|
| VXSC* | PDVL*  | #    | J=(UA)(UB1) | J=(UA)(UB2) | J=(UA)(UB3) |
|       | STARAD | +6,2 |             |             |             |

|        |        |        |               |             |             |
|--------|--------|--------|---------------|-------------|-------------|
| VXSC*  | STARAD | +12D,2 |               |             |             |
| STOVL* | 24D    |        | # K=(VA)(VB1) | J=(VA)(VB2) | J=(VA)(VB3) |
|        | 12D,1  |        |               |             |             |

|       |        |        |   |             |             |             |
|-------|--------|--------|---|-------------|-------------|-------------|
| VXSC* | VAD    |        |   |             |             |             |
|       | STARAD | +18D,2 | # | L=(WA)(WB1) | J=(WA)(WB2) | J=(WA)(WB3) |
| VAD   | VSL1   |        |   |             |             |             |

|         |                    |
|---------|--------------------|
| XCHX, 1 | 24D<br>UNIT<br>30D |
|---------|--------------------|

|       |     |        |               |             |             |
|-------|-----|--------|---------------|-------------|-------------|
| STORE | XDC | +18D,1 | # XDC = L+J+K | YDC = L+J+K | ZDC = L+J+K |
| TIX,1 |     |        |               |             |             |

AXISGEN3

AXISGEN3                      TIX,2

AXISGEN2

VLOAD

|       |                      |
|-------|----------------------|
| STOVL | XDC<br>STARAD<br>YDC |
|-------|----------------------|

|       |        |      |
|-------|--------|------|
| STOVL | STARAD | +6   |
|       | ZDC    |      |
| STORE | STARAD | +12D |

RVQ

|         |      |             |
|---------|------|-------------|
| QTSN45  | 2DEC | .1768       |
| .166... | 2DEC | .1666666667 |



# INFLIGHT\_ALIGNMENT\_ROUTINES

PAGE 1258 (EMPTY PAGE)

1412THE

|    |  |    |
|----|--|----|
| 1  |  | 1  |
| 2  |  | 2  |
| 3  |  | 3  |
| 4  |  | 4  |
| 5  |  | 5  |
| 6  |  | 6  |
| 7  |  | 7  |
| 8  |  | 8  |
| 9  |  | 9  |
| 10 |  | 10 |
| 11 |  | 11 |
| 12 |  | 12 |
| 13 |  | 13 |
| 14 |  | 14 |
| 15 |  | 15 |
| 16 |  | 16 |
| 17 |  | 17 |
| 18 |  | 18 |
| 19 |  | 19 |
| 20 |  | 20 |
| 21 |  | 21 |
| 22 |  | 22 |
| 23 |  | 23 |
| 24 |  | 24 |
| 25 |  | 25 |
| 26 |  | 26 |
| 27 |  | 27 |
| 28 |  | 28 |
| 29 |  | 29 |
| 30 |  | 30 |
| 31 |  | 31 |
| 32 |  | 32 |
| 33 |  | 33 |
| 34 |  | 34 |
| 35 |  | 35 |
| 36 |  | 36 |
| 37 |  | 37 |
| 38 |  | 38 |
| 39 |  | 39 |
| 40 |  | 40 |
| 41 |  | 41 |
| 42 |  | 42 |
| 43 |  | 43 |
| 44 |  | 44 |
| 45 |  | 45 |
| 46 |  | 46 |
| 47 |  | 47 |
| 48 |  | 48 |
| 49 |  | 49 |
| 50 |  | 50 |
| 51 |  | 51 |
| 52 |  | 52 |
| 53 |  | 53 |
| 54 |  | 54 |
| 55 |  | 55 |
| 56 |  | 56 |
| 57 |  | 57 |
| 58 |  | 58 |
| 59 |  | 59 |
| 60 |  | 60 |

```
1 BANK 14 # SAME FBANK AS THE FINDCDUD SUB-PROGRAM
2 SETLOC POWFLITE
3 BANK
```

```
4
5
6 EBANK= DEXDEX
7 COUNT* $$/POWFL
```

```
8
9 # CDUTRIG, CDUTRIG1, CDUTRIG2, AND CD*TR*GS ALL COMPUTE THE SINES AND
10 # COSINES OF THREE 2'S COMPLEMENT ANGLES AND PLACE THE RESULT, DOUBLE
11 # PRECISION, IN THE SAME ORDER AS THE INPUTS, AT SINCDU AND COSCDU. AN
12 # ADDITIONAL OUTPUT IS THE 1'S COMPLEMENT ANGLES AT CDUSPOT. THESE
13 # ROUTINES GO OUT OF THEIR WAY TO LEAVE THE MPAC AREA AS THEY FIND IT.
14 # EXCEPT FOR THE GENERALLY UNIMPORTANT MPAC +2. THEY DIFFER ONLY IN
15 # WHERE THEY GET THE ANGLES, AND IN METHOD OF CALLING.
```

```
16 #
17 # CDUTRIG (AND CDUTRIG1, WHICH CAN BE CALLED IN BASIC) COMPUTE THE
18 # SINES AND COSINES FROM THE CURRENT CONTENTS OF THE CDU REGISTERS.
19 # THE CONTENTS OF CDUTEMP, ETC., ARE NOT TOUCHED SO THAT THEY MAY
20 # CONTINUE TO FORM A CONSISTENT SET WITH THE LATEST PIPA READINGS.
```

```
21 #
22 # CDUTRIG1 IS LIKE CDUTRIG EXCEPT THAT IT CAN BE CALLED IN BASIC.
```

```
23 #
24 # CD*TR*GS FINDS CDU VALUES IN CDUSPOT RATHER THAN IN CDUTEMP. THIS
25 # ALLOWS USERS TO MAKE TRANSFORMATIONS USING ARBITRARY ANGLES, OR REAL
26 # ANGLES IN AN ORDER OTHER THAN X Y Z. A CALL TO THIS ROUTINE IS
27 # NECESSARY IN PREPARATION FOR A CALL TO AX*SR*T IN EITHER OF ITS TWO
28 # MODES (SMNB OR NBSM). SINCE AX*SR*T EXPECTS TO FIND THE SINES AND
29 # COSINES IN THE ORDER Y Z X THE ANGLES MUST HAVE BEEN PLACED IN CDUSPOT
30 # IN THIS ORDER. CD*TR*GS NEED NOT BE REPEATED WHEN AX*SR*T IS CALLED
31 # MORE THAN ONCE, PROVIDED THE ANGLES HAVE NOT CHANGED. NOTE THAT SINCE
32 # IT CLOBBERS BUF2 (IN THE SINE AND COSINE ROUTINES) CD*TR*GS CANNOT BE
33 # CALLED USING BANKCALL. SORRY.
```

```
34 #
35 # CD*TR*G IS LIKE CD*TR*GS EXCEPT THAT IT CAN BE CALLED IN
36 # INTERPRETIVE.
```

```
37
38 CDUTRIG EXIT
39 TC CDUTRIGS
40 TC INTPRET
41 RVQ
```

```
42
43 CD*TR*G EXIT
44 TC CD*TR*GS
45 TC INTPRET
46 RVQ
```

```
47
48 CDUTRIGS CA CDUX
49 TS CDUSPOT +4
50 CA CDUY
51 TS CDUSPOT
```

14121HE 1

# \*\*\*\*\*  
# QUICTRIG, INTENDED FOR GUIDANCE CYCLE USE WHERE TIME IS CRITICAL, IS A MUCH FASTER VERSION OF CD\*TR\*GS.  
# QUICTRIG COMPUTES AND STORES THE SINES AND COSINES OF THE 2'S COMPLEMENT ANGLES AT CDUSPOT, CDUSPOT +2,  
# AND CDUSPOT +4. UNLIKE CD\*TR\*GS, QUICTRIG DOES NOT LEAVE THE 1'S COMPLEMENT VERSIONS OF THE ANGLES IN  
# CDUSPOT. QUICTRIG'S EXECUTION TIME IS 4.1 MS; THIS IS 10 TIMES AS FAST AS CD\*TR\*GS. QUICTRIG MAY BE  
# CALLED FROM INTERPRETIVE AS AN RTB OP-CODE, OR FROM BASIC VIA BANKCALL OR IBNKCALL.

|          |        |                                        |
|----------|--------|----------------------------------------|
| QUICTRIG | INHINT | # INHINT SINCE DAP USES THE SAME TEMPS |
|          | EXTEND |                                        |
|          | QXCH   | ITEMP1                                 |
|          | CAF    | FOUR                                   |
| +4       | MASK   | SIX                                    |
|          | TS     | ITEMP2                                 |
|          | INDEX  | ITEMP2                                 |
|          | CA     | CDUSPOT                                |
|          | TC     | SPSIN                                  |
|          | EXTEND |                                        |
|          | MP     | BIT14                                  |
|          | INDEX  | ITEMP2                                 |
|          | DXCH   | SINCDU                                 |
|          | INDEX  | ITEMP2                                 |
|          | CA     | CDUSPOT                                |
|          | TC     | SPCOS                                  |
|          | EXTEND |                                        |
|          | MP     | BIT14                                  |
|          | INDEX  | ITEMP2                                 |
|          | DXCH   | COSCDU                                 |
|          | CCS    | ITEMP2                                 |
|          | TCF    | QUICTRIG +4                            |
|          | CA     | ITEMP1                                 |
|          | RELINT |                                        |
|          | TC     | A                                      |

```

THESE INTERFACE ROUTINES MAKE IT POSSIBLE TO CALL AX*SR*T, ETC., IN
INTERPRETIVE. LATER, WHERE POSSIBLE, THEY WILL BE ELIMINATED.
#
THESE INTERFACE ROUTINES ARE PERMANENT. ALL RESTORE USER'S EBANK
SETTING. ALL ARE STRICT INTERPRETIVE SUBROUTINES, CALLED USING "CALL",
RETURNING VIA QPRET. ALL EXPECT AND RETURN THE VECTOR TO BE TRANSFORMED
INTERPRETER-STYLE IN MPAC; COMPONENTS AT MPAC, MPAC +3, AND MPAC +5.
#
TRG*SMNB AND TRG*NBSM BOTH EXPECT TO SEE THE 2'S COMPLEMENT ANGLES
AT CDUSPOT (ORDER Y Z X, AT CDUSPOT, CDUSPOT +2, AND CDUSPOT +4; ODD
LOCATIONS NEED NOT BE ZEROED). TRG*NBSM DOES THE NB TO SM TRANSFORMATION;
TRG*SMNB, VICE VERSA.
#
CDU*NBSM DOES ITS TRANSFORMATION USING THE PRESENT CONTENTS OF
THE CDL COUNTERS. OTHERWISE IT IS LIKE TRG*NBSM.
#
CDU*SMNB IS THE COMPLEMENT OF CDU*NBSM.

CDU*SMNB EXIT
 TC CDUTRIGS
 TCF C*MM*N1

TRG*SMNB EXIT
 TC CD*TR*GS
C*MM*N1 TC MPACVBUF # AX*SR*T EXPECTS VECTOR IN VBUF
 CS THREE # SIGNAL FOR SM TO NB TRANSFORMATION.
C*MM*N2 TC AX*SR*T
 TC INTRET
 VLOAD RVQ
 VBUF

CDU*NBSM EXIT
 TC CDUTRIGS
 TCF C*MM*N3

TRG*NBSM EXIT
 TC CD*TR*GS
C*MM*N3 TC MPACVBUF # FOR AX*SR*T
 CA THREE # SIGNAL FOR NB TO SM TRANSFORMATION
 TCF C*MM*N2

NBSM AND *SMNB* EXPECT TO SEE THE SINES AND COSINES (AT SINCDU
AND COSCDU) RATHER THAN THE ANGLES THEMSELVES. OTHERWISE THEY ARE
LIKE TRG*NBSM AND TRG*SMNB.
#
NOTE THAT JUST AS CD*TR*GS NEED BE CALLED ONLY ONCE FOR EACH SERIES
OF TRANSFORMATIONS USING THE SAME ANGLES, SO TOO ONLY ONE OF TRG*NBSM
```

# AND TRG\*SMNB NEED BE CALLED FOR EACH SERIES. FOR SUBSEQUENT TRANFOR-  
# MATIONS USE \*NBSM\* AND \*SMNB\*.

\*SMNB\*           EXIT  
                  TCF       C\*MM\*N1

\*NBSM\*           EXIT  
                  TCF       C\*MM\*N3

# AX\*SR\*T COMBINES THE OLD SMNB AND NBSM. FOR THE NB TO SM  
# TRANSFORMATION, ENTER WITH +3 IN A. FOR SM TO NB, ENTER WITH -3.  
# THE VECTOR TO BE TRANSFORMED ARRIVES, AND IS RETURNED, IN VBUF.  
# AX\*SR\*T EXPECTS TO FIND THE SINES AND COSINES OF THE ANGLES OF ROTATION  
# AT SINCDU AND COSCDU, IN THE ORDER Y Z X. A CALL TO CD\*TR\*GS, WITH  
# THE 2'S COMPLEMENT ANGLES (ORDER Y Z X) AT CDUSPOT, WILL TAKE CARE OF  
# THIS. HERE IS A SAMPLE CALLING SEQUENCE:--

#               TC       CDUTRIGS  
#               CS       THREE           # ("CA THREE" FOR NBSM)  
#               TC       AX\*SR\*T

# THE CALL TO CD\*TR\*GS NEED NOT BE REPEATED, WHEN AX\*SR\*T IS CALLED MORE  
# THAN ONCE, UNLESS THE ANGLES HAVE CHANGED.

#  
# AX\*SR\*T IS GUARANTEED SAFE ONLY FOR VECTORS OF MAGNITUDE LESS THAN  
# UNITY. A LOOK AT THE CASE IN WHICH A VECTOR OF GREATER MAGNITUDE  
# HAPPENS TO LIE ALONG AN AXIS OF THE SYSTEM TO WHICH IT IS TO BE TRANS-  
# FORMED CONVINCES ONE THAT THIS IS A RESTRICTION WHICH MUST BE ACCEPTED.

AX\*SR\*T           TS       DEXDEX           # WHERE IT BECOMES THE INDEX OF INDEXES.  
                  EXTEND

                  QXCH    RTNSAVER

R\*TL\*\*P           CCS       DEXDEX       #               +3 --> 0       -3 --> 2  
                  CS       DEXDEX       # THUS:       +2 --> 1       -2 --> 1  
                  AD       THREE       #               +1 --> 2       -1 --> 0

                  EXTEND  
                  INDEX    A  
                  DCA       INDEXI  
                  DXCH     DEXI

                  CA       ONE  
                  TS       BUF

                  EXTEND  
                  INDEX    DEX1  
                  DCS       VBUF

                  TCF       LOOP1       # REALLY BE A SUBTRACT, AND VICE VERSA

LOOP2            DXCH    BUF           # LOADING VECTOR COMPONENT, STORING INDEX



|    |          |        |          |                                           |       |       |    |
|----|----------|--------|----------|-------------------------------------------|-------|-------|----|
| 2  | LOOP1    | DXCH   | MPAC     |                                           |       |       | 2  |
| 3  |          | CA     | SINSLOC  |                                           |       |       | 3  |
| 4  |          | AD     | DEX1     |                                           |       |       | 4  |
| 5  |          | TS     | ADDRWD   |                                           |       |       | 5  |
| 6  |          |        |          |                                           |       |       | 6  |
| 7  |          | TC     | DMPSUB   | # MULTIPLY AT SIN(CDUANGLE)               |       |       | 7  |
| 8  |          | CCS    | DEXDEX   |                                           |       |       | 8  |
| 9  |          | DXCH   | MPAC     | # NBSM CASE                               |       |       | 9  |
| 10 |          | TCF    | +3       |                                           |       |       | 10 |
| 11 |          | EXTEND |          | # SMNB CASE                               |       |       | 11 |
| 12 |          | DCS    | MPAC     |                                           |       |       | 12 |
| 13 |          | DXCH   | TERM1TMP |                                           |       |       | 13 |
| 14 |          |        |          |                                           |       |       | 14 |
| 15 |          | CA     | SIX      | # SINCDU AND COSCDU (EACH 6 WORDS) MUST   |       |       | 15 |
| 16 |          | ADS    | ADDRWD   | # BE CONSECUTIVE AND IN THAT ORDER        |       |       | 16 |
| 17 |          |        |          |                                           |       |       | 17 |
| 18 |          | EXTEND |          |                                           |       |       | 18 |
| 19 |          | INDEX  | BUF      |                                           |       |       | 19 |
| 20 |          | INDEX  | DEX1     |                                           |       |       | 20 |
| 21 |          | DCA    | VBUF     |                                           |       |       | 21 |
| 22 |          | DXCH   | MPAC     |                                           |       |       | 22 |
| 23 |          | TC     | DMPSUB   | # MULTIPLY BY COS(CDUANGLE)               |       |       | 23 |
| 24 |          | DXCH   | MPAC     |                                           |       |       | 24 |
| 25 |          | DAS    | TERM1TMP |                                           |       |       | 25 |
| 26 |          | DXCH   | TERM1TMP |                                           |       |       | 26 |
| 27 |          | DDOUBL |          |                                           |       |       | 27 |
| 28 |          | INDEX  | BUF      |                                           |       |       | 28 |
| 29 |          | INDEX  | DEX1     |                                           |       |       | 29 |
| 30 |          | DXCH   | VBUF     |                                           |       |       | 30 |
| 31 |          | DXCH   | BUF      | # LOADING INDEX, STORING VECTOR COMPONENT |       |       | 31 |
| 32 |          |        |          |                                           |       |       | 32 |
| 33 |          | CCS    | A        | # 'CAUSE THAT'S WHERE THE INDEX NOW IS    |       |       | 33 |
| 34 |          | TCF    | LOOP2    |                                           |       |       | 34 |
| 35 |          |        |          |                                           |       |       | 35 |
| 36 |          | EXTEND |          |                                           |       |       | 36 |
| 37 |          | DIM    | DEXDEX   | # DECREMENT MAGNITUDE PRESERVING SIGN     |       |       | 37 |
| 38 |          |        |          |                                           |       |       | 38 |
| 39 | TSTPOINT | CCS    | DEXDEX   | # ONLY THE BRANCHING FUNCTION IS USED     |       |       | 39 |
| 40 |          | TCF    | R*TL**P  |                                           |       |       | 40 |
| 41 |          | TC     | RTNSAVER |                                           |       |       | 41 |
| 42 |          | TCF    | R*TL**P  |                                           |       |       | 42 |
| 43 |          | TC     | RTNSAVER |                                           |       |       | 43 |
| 44 |          |        |          |                                           |       |       | 44 |
| 45 | SINSLOC  | ADRES  | SINCDU   | # FOR USE IN SETTING ADDRWD               |       |       | 45 |
| 46 |          |        |          |                                           |       |       | 46 |
| 47 | INDEXI   | DEC    | 4        | # *****                                   | DON'T | ***** | 47 |
| 48 |          | DEC    | 2        | # *****                                   | TOUCH | ***** | 48 |
| 49 |          | DEC    | 0        | # *****                                   | THESE | ***** | 49 |
| 50 |          |        |          |                                           |       |       | 50 |
| 51 |          |        |          |                                           |       |       | 51 |
| 52 |          |        |          |                                           |       |       | 52 |
| 53 |          |        |          |                                           |       |       | 53 |
| 54 |          |        |          |                                           |       |       | 54 |
| 55 |          |        |          |                                           |       |       | 55 |
| 56 |          |        |          |                                           |       |       | 56 |
| 57 |          |        |          |                                           |       |       | 57 |
| 58 |          |        |          |                                           |       |       | 58 |
| 59 |          |        |          |                                           |       |       | 59 |
| 60 |          |        |          |                                           |       |       | 60 |

```
DEC 4 # ***** CONSTANTS *****
```

```

```

```
BANK 10
SETLOC FLESHLOC
BANK
COUNT* $$/POWFL
```

```
ROUTINE FLESHPOT COMPUTES THE BODY-STABLE MEMBER TRANSFORMATION MATRIX (COMMONLY CALLED XNB) AND STORES
IT IN THE LOCATIONS SPECIFIED BY THE ECADR ENTERING IN A.
```

```
CALCSMSC EXIT
 TC BANKCALL
 CADR FLESHPOT -1
 TC INTPRET
 RVQ
```

```
XNBECADR ECADR XNB
```

```
-1 CAF XNBECADR
```

```
FLESHPOT TS TEM2
 XCH EBANK
 XCH TEM2
 MASK LOW8
 AD OCT1400
 TS TEM1
```

```
EXTEND
DCA COSCDUY
DXCH MPAC
TC DMP
ADRES COSCDUZ
DXCH MPAC
```

```
DDOUBL
INDEX TEM1
DXCH 0 # = COSY COSZ
```

```
EXTEND
DCA SINCDUZ
INDEX TEM1
DXCH 2 # = SINZ
```

```
EXTEND
DCS SINCDUY
DXCH MPAC
TC DMPSUB # ADDRWD SET TO COSCDUZ
```

[illegible]

|    |  |        |          |                                  |    |
|----|--|--------|----------|----------------------------------|----|
| 1  |  |        |          |                                  | 1  |
| 2  |  | DDOUBL |          |                                  | 2  |
| 3  |  | DAS    | BUF      |                                  | 3  |
| 4  |  |        |          |                                  | 4  |
| 5  |  | DXCH   | BUF      |                                  | 5  |
| 6  |  | DXCH   | MPAC     |                                  | 6  |
| 7  |  |        |          |                                  | 7  |
| 8  |  | EXTEND |          |                                  | 8  |
| 9  |  | DCA    | MPAC     |                                  | 9  |
| 10 |  | INDEX  | TEM1     |                                  | 10 |
| 11 |  | DXCH   | 14       | # = - SINY COSX + SINX SINZ COSY | 11 |
| 12 |  |        |          |                                  | 12 |
| 13 |  | EXTEND |          |                                  | 13 |
| 14 |  | DCA    | MPAC     | +3                               | 14 |
| 15 |  | INDEX  | TEM1     |                                  | 15 |
| 16 |  | DXCH   | 16       | # = - SINX COSZ                  | 16 |
| 17 |  |        |          |                                  | 17 |
| 18 |  | EXTEND |          |                                  | 18 |
| 19 |  | DCA    | MPAC     | +5                               | 19 |
| 20 |  | INDEX  | TEM1     |                                  | 20 |
| 21 |  | DXCH   | 20       | # = COSX COSY - SINX SINY SINZ   | 21 |
| 22 |  |        |          |                                  | 22 |
| 23 |  | CA     | TEM1     |                                  | 23 |
| 24 |  | TS     | ADDRWD   |                                  | 24 |
| 25 |  | EXTEND |          |                                  | 25 |
| 26 |  | DCA    | Z        |                                  | 26 |
| 27 |  | AD     | FOUR     |                                  | 27 |
| 28 |  | DXCH   | LOC      |                                  | 28 |
| 29 |  | CAF    | BIT8     |                                  | 29 |
| 30 |  | TS     | EDOP     |                                  | 30 |
| 31 |  | TCF    | VXV      |                                  | 31 |
| 32 |  | DXCH   | MPAC     |                                  | 32 |
| 33 |  | DDOUBL |          |                                  | 33 |
| 34 |  | INDEX  | TEM1     |                                  | 34 |
| 35 |  | DXCH   | 6        |                                  | 35 |
| 36 |  |        |          |                                  | 36 |
| 37 |  | DXCH   | MPAC     | +3                               | 37 |
| 38 |  | DDOUBL |          |                                  | 38 |
| 39 |  | INDEX  | TEM1     |                                  | 39 |
| 40 |  | DXCH   | 10       |                                  | 40 |
| 41 |  |        |          |                                  | 41 |
| 42 |  | DXCH   | MPAC     | +5                               | 42 |
| 43 |  | DDOUBL |          |                                  | 43 |
| 44 |  | INDEX  | TEM1     |                                  | 44 |
| 45 |  | DXCH   | 12       |                                  | 45 |
| 46 |  |        |          |                                  | 46 |
| 47 |  | CA     | TEM2     |                                  | 47 |
| 48 |  | TS     | EBANK    |                                  | 48 |
| 49 |  | TCF    | SWRETURN |                                  | 49 |
| 50 |  |        |          |                                  | 50 |
| 51 |  |        |          |                                  | 51 |
| 52 |  |        |          |                                  | 52 |
| 53 |  |        |          |                                  | 53 |
| 54 |  |        |          |                                  | 54 |
| 55 |  |        |          |                                  | 55 |
| 56 |  |        |          |                                  | 56 |
| 57 |  |        |          |                                  | 57 |
| 58 |  |        |          |                                  | 58 |
| 59 |  |        |          |                                  | 59 |
| 60 |  |        |          |                                  | 60 |

```
1 # THE TFF SUBROUTINES MAY BE USED IN EITHER EARTH OR MOON CENTERED COORDINATES. THE TFF ROUTINES NEVER
2 # KNOW WHICH ORIGIN APPLIES. IT IS THE USER WHO KNOWS, AND WHO SUPPLIES RONE, VONE, AND 1/SQRT(MU) AT THE
3 # APPROPRIATE SCALE LEVEL FOR THE PROPER PRIMARY BODY.
4 #
5 #
6 # EARTH ORIGIN POSITION -29 METERS
7 # VELOCITY -7 METERS/CENTISECOND
8 # 1/SQRT(MU) +17 SQRT(CS SQ/METERS CUBED)
9 #
10 # MOON ORIGIN POSITION -27 METERS
11 # VELOCITY -5 METERS/CENTISECONDS
12 # 1/SQRT(MU) +14 SQRT(CS SQ/METERS CUBED)
13 #
14 # ALL DATA PROVIDED TO AND RECEIVED FROM ANY TFF SUBROUTINE WILL BE AT ONE OF THE LEVELS ABOVE. IN ALL CASES,
15 # THE FREE FALL TIME IS RETURNED IN CENTISECONDS AT (-28). PROGRAM TFF/CONIC WILL GENERATE VONE/RTMU AND
16 # LEAVE IT IN VONE' AT (+10) IF EARTH ORIGIN AND (+9) IF MOON ORIGIN.
17 #
18 # THE USER MUST STORE THE STATE VECTOR IN RONE, VONE, AND MU IN THE FORM 1/SQRT(MU) IN TFF/RTMU
19 # AT THE PROPER SCALE BEFORE CALLING TFF/CONIC. SINCE RONE, VONE ARE IN THE EXTENDED VERB STORAGE AREA,
20 # THE USER MUST ALSO LOCK OUT THE EXTENDED VERBS, AND RELEASE THEM WHEN FINISHED.
21 #
22 # PROGRAMS CALC/TFF AND CALC/TPER ASSUME THAT THE TERMINAL RADIUS IS LESS THAN THE PRESENT
23 # RADIUS. THIS RESTRICTION CAN BE REMOVED BY A 15 W CODING CHANGE, BUT AT PRESENT IT IS NOT DEEMED NECESSARY.
24 #
25 # THE FOLLOWING ERASABLE QUANTITIES ARE USED BY THE TFF ROUTINES, AND ARE LOCATED IN THE PUSH LIST.
26 #
27 # BELOW E: IS USED FOR EARTH ORIGIN SCALE
28 # M: IS USED FOR MOON ORIGIN SCALE
29 #
30 #TFFSW = 119D # BIT1 0 = CALCTFF 1 = CALCTPER
31 TFFDELQ = 10D # Q2-Q1 E: (-16) M: (-15)
32 RMAG1 = 12D # ABVAL(RN) M E: (-29) M: (-27)
33 #RPER = 14D # PERIGEE RADIUS M E: (-29) M: (-27)
34 TFFQ1 = 14D # R.V / SQRT(MUE) E: (-16) M: (-15)
35 #SDELF/2 # SIN(THETA) /2
36 CDELF/2 = 14D # COS(THETA) /2
37 #RAPO = 16D # APOGEE RADIUS M E: (-29) M: (-27)
38 NRTERM = 16D # TERMINAL RADIUS M E: (-29+NR)
39 # M: (-27+NR)
40 RTERM = 18D # TERMINAL RADIUS M E: (-29) M: (-27)
41 TFFVSQ = 20D # -(V SQUARED/MU) 1/M E: (20) M: (18)
42 TFF1/ALF = 22D # SEMI MAJ AXIS M E: (-22-2 NA)
43 # M: (-20-2 NA)
44 TFFRTALF = 24D # SQRT(ALFA) E: (10+NA) M: (9+NA)
45 TFFALFA = 26D # ALFA 1/M E: (26-NR) M: (24-NR)
46 TFFNP = 28D # SEMI LATUS RECTUM M E: (-38+2 NR)
47 # M: (-36+2 NR)
48 TFF/RTMU = 30D # 1/SQRT(MU) E: (17) M: (14)
49 NRMAG = 32D # PRESENT RADIUS M E: (-29+NR)
50 # M: (-27+NR)
51 TFFX = 34D #
52 TFFTEM = 36D # TEMPORARY
```

|   |                                                                         |
|---|-------------------------------------------------------------------------|
| # | REGISTERS S1, S2 ARE UNTOUCHED BY ANY TFF SUBROUTINE                    |
| # | INDEX REGISTERS X1, X2 ARE USED BY ALL TFF SUBROUTINES. THEY ARE ESTAB- |
| # | LISHED IN TFF/CONIC AND MUST BE PRESERVED BETWEEN CALLS TO SUBSEQUENT   |
| # | SUBROUTINES.                                                            |
| # | -NR C(X1) = NORM COUNT OF RMAG                                          |
| # | -NA C(X2) = NORM COUNT OF SQRT(ABS(ALFA))                               |

```
1 # SUBROUTINE NAME: TFFCONIC DATE: 01.29.67
2 # MOD NO: 0 LOG SECTION: TIME OF FREE FALL
3
4 # MOD BY: RR BAIRNSFATHER
5 # MOD NO: 1 MOD BY: RR BAIRNSFATHER DATE: 11 APR 67
6 # MOD NO: 2 MOD BY: RR BAIRNSFATHER DATE: 21 NOV 67 ADD MOON MU.
7 # MOD NO: 3 MOD BY: RR BAIRNSFATHER DATE: 21 MAR 68 ACCEPT DIFFERENT EARTH/MOON SCALES
8 #
9 # FUNCTIONAL DESCRIPTION: THIS SUBROUTINE IS CALLED TO COMPUTE THOSE CONIC PARAMETERS REQUIRED BY THE TFF
10 # SUBROUTINES AND TO ESTABLISH THEM IN THE PUSH LIST AREA. THE PARAMETERS ARE LISTED UNDER OUTPUT.
11 # THE EQUATIONS ARE:
12 #
13 # $\bar{H} = \bar{R}\bar{N} * \bar{V}\bar{N}$ ANGULAR MOMENTUM
14 #
15 # $LCP = \bar{H} . \bar{H} / MU$ SEMI LATUS RECTUM
16 #
17 # $ALFA = 2/RN - \bar{V}\bar{N} . \bar{V}\bar{N} / MU$ RECIPROCAL SEMI-MAJOR AXIS, SIGNED
18 #
19 # AND ALFA IS POS FOR ELLIPTIC ORBITS
20 # 0 FOR PARABOLIC ORBITS
21 # NEG FOR HYPERBOLIC ORBITS
22 # SUBROUTINE ALSO COMPUTES AND SAVES RMAG.
23 #
24 # CALLING SEQUENCE:
25 # TFFCONIC EXPECTS CALLER TO ENTER WITH CORRECT GRAVITATIONAL CONSTANT IN MPAC, IN THE FORM
26 # 1/SQRT(MU). THE PROGRAM WILL SAVE IN TFF/RTMU. THE SCALE IS DETERMINED BY WHETHER EARTH OR MOON
27 # ORIGIN IS USED. THE CALLER MUST LOCK OUT THE EXTENDED VERBS BEFORE PROVIDING STATE VECTOR IN RONE,
28 # VONE AT PROPER SCALE. THE EXTENDED VERBS MUST BE RESTORED WHEN THE CALLER IS FINISHED USING THE
29 # TFF ROUTINES.
30 #
31 # ENTRY POINT TFFCONMU EXPECTS THAT TFF/RTMU IS ALREADY LOADED.
32 #
33 # TO SPECIFY MU: DLOAD CALL IF MU ALREADY STORED: CALL
34 # YOURMU 1/RTMU E:(17) M:(14) TFFCONMU
35 # TFFCONIC
36 # PUSHLOC = PDL+0, ARBITRARY IF LEQ 18D
37 #
38 # SUBROUTINES CALLED: NONE
39 #
40 # NORMAL EXIT MODES: RVQ
41 #
42 # ALARMS: NONE
43 #
44 # OUTPUT: THE FOLLOWING ARE STORED IN THE PUSH LIST AREA.
45 # RMAG1 E:(-29) M:(-27) M RN, PRESENT RADIUS LENGTH.
46 # NRMAG E:(-29+NR) M RMAG, NORMALIZED
47 # M:(-27+NR)
48 # X1 -NR, NORM COUNT
49 # TFFNP E:(-38+2NR) M LCP, SEMI LATUS RECTUM, WEIGHTED BY NR. FOR VGAMCALC.
50 # M:(-36+2NR)
51 # TFF/RTMU E:(17) M:(14) 1/SQRT(MU)
52 # TFFVSQ E:(20) M:(18) 1/M -(V SQ/MU): PRESENT VELOCITY, NORMALIZED. FOR VGAMCALC
53 # TFFALFA E:(26-NR) 1/M ALFA, WEIGHTED BY NR
54 # M:(24-NR)
55 # TFFRTALF E:(10+NA) SQRT(ALFA), NORMALIZED
56 # M:(9+NA)
```

```
1 # X2 -NA, NORMCOUNT
2 # TFF1/ALF E:(-22-2NA) SIGNED SEMI MAJ AXIS, WEIGHTED BY NA
3 # M:(-20-2NA)
4 # PUSHLOC AT PDL+0
5 #
6 # THE FOLLOWING IS STORED IN GENERAL ERASABLE
7 # VONE' E:(10) M:(9) V/RT(MU), NORMALIZED VELOCITY
8 #
9 # ERASABLE INITIALIZATION REQUIRED:
10 # RONE E:(-29) M:(-27) M STATE VECTOR LEFT BY CALLER
11 # VONE E:(-7) M:(-5) M/CS STATE VECTOR LEFT BY CALLER
12 # TFF/RTMU E:(17) M:(14) 1/RT(CS SQ/M CUBE) IF ENTER VIA TFFCONMU.
13 #
14 # DEBRIS: QPRET PDL+0 ... PDL+3
15
16 BANK 33
17 SETLOC TOF-FF
18 BANK
19
20 COUNT* $$/TFF
21
22 TFFCONIC STORE TFF/RTMU # 1/SQRT(MU) E:(17) M:(14)
23
24 TFFCONMU VLOAD UNIT # COME HERE WITH TFFRTMU LOADED.
25 RONE # SAVED RN. M E:(-29) M:(-27)
26 PDDL # UR/2 TO PDL+0, +5
27
28 # MAGNITUDE
29 STORE 36D # M E:(-29) M:(-27)
30 RMAG1
31
32 NORM
33
34 X1 # -NR
35 STOVL NRMAG # RMAG M E:(-29+NR) M:(-27+NR)
36 VONE # SAVED VN. M/CS E:(-7) M:(-5)
37
38 VXSC
39 TFF/RTMU # E:(17) M:(14)
40 STORE VONE' # VN/SQRT(MU) E:(10) M:(9)
41
42 VXSC VXV
43 NRMAG # E:(-29+NR) M:(-27+NR)
44 # UR/2 FROM PDL
45 VSL1 VSQ # BEFORE: E:(-19+NR) M:(-18+NR)
46 STODL TFFNP # LC P M E:(-38+2NR) M:(-36+2NR)
47 # SAVE ALSO FOR VGAMCALC
48 TFF1/4
49 DDV PDVL # (2/RMAG) 1/M E:(26-NR) M:(24-NR)
50 NRMAG # RMAG M E:(-29+NR) M:(-27+NR)
51 VONE' # SAVED VN. E:(10) M:(9)
52
53 VSQ DCOMP # KEEP MPAC+2 HONEST FOR SQRT.
54 STORE TFFVSQ # -(V SQ/MU) E:(20) M:(18)
55 # SAVE FOR VGAMCALC
56
57 SR* DAD
```





|    |          |       |          |                                        |    |
|----|----------|-------|----------|----------------------------------------|----|
| 1  |          |       |          |                                        | 1  |
| 2  |          | 0     | -6,1     | # GET -VSQ/MU E:(26-NR) M:(24-NR)      | 2  |
| 3  |          | STADR |          |                                        | 3  |
| 4  |          |       |          | # 2/RMAG FROM PDL+2                    | 4  |
| 5  |          | STORE | TFFALFA  | # ALFA 1/M E:(26-NR) M:(24-NR)         | 5  |
| 6  |          | SL*   | PUSH     | # TEMP SAVE ALFA E:(20) M:(18)         | 6  |
| 7  |          |       | 0        |                                        | 7  |
| 8  |          | ABS   | SQRT     | # E:(10) M:(9)                         | 8  |
| 9  |          | NORM  |          |                                        | 9  |
| 10 |          |       | X2       | # X2 = -NA                             | 10 |
| 11 |          | STORE | TFFRTALF | # SQRT( ABS(ALFA) ) E:(10+NA) M:(9+NA) | 11 |
| 12 |          | DSQ   | SIGN     | # NOT SO ACCURATE, BUT OK              | 12 |
| 13 |          |       |          | # ALFA FROM PDL+2 E:(20) M:(18)        | 13 |
| 14 |          | BZE   | BDDV     | # SET 1/ALFA =0, TO SHOW SMALL ALFA    | 14 |
| 15 |          |       | +2       |                                        | 15 |
| 16 |          |       | TFF1/4   |                                        | 16 |
| 17 | +2       | STORE | TFF1/ALF | # 1/ALFA E:(-22-2NA) M:(-20-2NA)       | 17 |
| 18 | DUMPCNIC | RVQ   |          |                                        | 18 |
| 19 |          |       |          |                                        | 19 |
| 20 | #        |       |          | 39 W                                   | 20 |
| 21 |          |       |          |                                        | 21 |
| 22 |          |       |          |                                        | 22 |
| 23 |          |       |          |                                        | 23 |
| 24 |          |       |          |                                        | 24 |
| 25 |          |       |          |                                        | 25 |
| 26 |          |       |          |                                        | 26 |
| 27 |          |       |          |                                        | 27 |
| 28 |          |       |          |                                        | 28 |
| 29 |          |       |          |                                        | 29 |
| 30 |          |       |          |                                        | 30 |
| 31 |          |       |          |                                        | 31 |
| 32 |          |       |          |                                        | 32 |
| 33 |          |       |          |                                        | 33 |
| 34 |          |       |          |                                        | 34 |
| 35 |          |       |          |                                        | 35 |
| 36 |          |       |          |                                        | 36 |
| 37 |          |       |          |                                        | 37 |
| 38 |          |       |          |                                        | 38 |
| 39 |          |       |          |                                        | 39 |
| 40 |          |       |          |                                        | 40 |
| 41 |          |       |          |                                        | 41 |
| 42 |          |       |          |                                        | 42 |
| 43 |          |       |          |                                        | 43 |
| 44 |          |       |          |                                        | 44 |
| 45 |          |       |          |                                        | 45 |
| 46 |          |       |          |                                        | 46 |
| 47 |          |       |          |                                        | 47 |
| 48 |          |       |          |                                        | 48 |
| 49 |          |       |          |                                        | 49 |
| 50 |          |       |          |                                        | 50 |
| 51 |          |       |          |                                        | 51 |
| 52 |          |       |          |                                        | 52 |
| 53 |          |       |          |                                        | 53 |
| 54 |          |       |          |                                        | 54 |
| 55 |          |       |          |                                        | 55 |
| 56 |          |       |          |                                        | 56 |
| 57 |          |       |          |                                        | 57 |
| 58 |          |       |          |                                        | 58 |
| 59 |          |       |          |                                        | 59 |
| 60 |          |       |          |                                        | 60 |

```
1 # SUBROUTINE NAME: TFFRP/RA DATE: 01.17.67
2 # MOD NO: 0 LOG SECTION: TIME OF FREE FALL
3
4 # MOD NO: 1 MOD BY: RR BAIRNSFATHER DATE: 11 APR 67
5 # MOD NO: 2 MOD BY: RR BAIRNSFATHER DATE: 21 MAR 68 ACCEPT DIFFERENT EARTH/MOON SCALES
6 # ALSO IMPROVE ACCURACY OF RAPO.
7
8 # FUNCTIONAL DESCRIPTION: USED BY CALCTPER AND TFF DISPLAYS TO CALCULATE PERIGEE RADIUS AND ALSO
9 # APOGEE RADIUS FOR A GENERAL CONIC.
10 # PROGRAM GIVES PERIGEE RADIUS AS APOGEE RADIUS IS GIVEN BY
11 # RP = P/(1+E) RA = (1+E) / ALFA
12 # WHERE 2
13 # E = 1 - P ALFA
14 # IF RA IS NEGATIVE OR SHOWS DIVIDE OVERFLOW, THEN RA = POSMAX BECAUSE
15 # 1. APOGEE RADIUS IS NOT MEANINGFUL FOR HYPERBOLA
16 # 2. APOGEE RADIUS IS NOT DEFINED FOR PARABOLA
17 # 3. APOGEE RADIUS EXCEEDS THE SCALING FOR ELLIPSE.
18
19 # THIS SUBROUTINE REQUIRED THE SIGNED RECIPROCAL SEMI MAJ AXIS, ALFA, AND SEMI-LATUS RECTUM AS DATA.
20 #
21 # CALLING SEQUENCE: CALL
22 # TFFRP/RA
23 # PUSHLOC = PDL+0, ARBITRARY IF LEQ 10D
24 # C(MPAC) UNSPECIFIED
25
26 # SUBROUTINES CALLED: NONE
27 #
28 # NORMAL EXIT MODE: RVQ
29 # IF ELLIPSE, WITHIN NORMAL SCALING, RAPO IS CORRECT.
30 # OTHERWISE, RAPO = POSMAX.
31
32 # ALARMS: NONE
33 #
34 # OUTPUT: STORED IN PUSH LIST AREA. SCALE OF OUTPUT AGREES WITH DATA SUPPLIED TO TFF/CONIC.
35 # RPER E:(-29) M:(-27) M PERIGEE RADIUS DESTROYED BY CALCTFF/CALCTPER, TFFTRIG.
36 # RAPO E:(-29) M:(-27) M APOGEE RADIUS WILL BE DESTROYED BY CALCTFF/CALCTPER
37 # PUSHLOC AT PDL+0
38 #
39 # ERASABLE INITIALIZATION REQUIRED:
40 # TFFALFA E:(26-NR) M 1/SEMI MAJ AXIS LEFT BY TFFCONIC
41 # M:(24-NR)
42 # TFFNP E:(-38+2NR) M LC P, SEMI LATUS RECTUM LEFT BY TFFCONIC
43 # M:(-36+2NR)
44 # X1 -NR, NORM COUNT OF RMAG LEFT BY TFFCONIC
45 # X2 -NA, NORM COUNT OF ALFA LEFT BY TFFCONIC
46 #
47 # DEBRIS: QPRET, PDL+0 ... PDL+1
```

|          |       |           |                                           |
|----------|-------|-----------|-------------------------------------------|
| RAPO     | =     | 16D       | # APOGEE RADIUS M E:(-29) M:(-27)         |
| RPER     | =     | 14D       | # PERIGEE RADIUS M E:(-29) M:(-27)        |
| TFFRP/RA | DLOAD | DMP       |                                           |
|          |       | TFFALFA   | # ALFA 1/M E:(26-NR) M:(24-NR)            |
|          |       | TFFNP     | # LC P M E:(-38+2NR) M:(-36+2NR)          |
|          | SR*   | DCOMP     | # ALFA P (-12+NR)                         |
|          |       | 0 -8D,1   | # ALFA P (-4)                             |
|          | DAD   | ABS       | # (DCOMP GIVES VALID TP RESULT FOR SQRT)  |
|          |       |           | # (ABS PROTECTS SQRT IF E IS VERY NEAR 0) |
|          |       | DP2(-4)   |                                           |
|          | SQRT  | DAD       | # E SQ = (1- P ALFA) (-4)                 |
|          |       | TFF1/4    |                                           |
|          | PUSH  | BDDV      | # (1+E) (-2) TO PDL+0                     |
|          |       | TFFNP     | # LCP M E:(-38+2NR) M:(-36+2NR)           |
|          | SR*   | SR*       | # (DOES SR THEN SL TO AVOID OVFL)         |
|          |       | 0,1       | # X1=-NR                                  |
|          |       | 0 -7,1    | # (EFFECTIVE SL)                          |
|          | STODL | RPER      | # PERIGEE RADIUS M E:(-29) M:(-27)        |
|          |       |           | # (1+E) (-2) FROM PDL+0                   |
|          | DMP   | BOVB      |                                           |
|          |       | TFF1/ALF  | # E:(-22-2NA) M:(-20-2NA)                 |
|          |       | TC DANZIG | # CLEAR OVFLND, IF ON.                    |
|          | BZE   | SL*       |                                           |
|          |       | MAXRA     | # SET POSMAX IF ALFA=0                    |
|          |       | 0 -5,2    | # -5+NA                                   |
|          | SL*   | BOV       |                                           |
|          |       | 0,2       |                                           |
|          |       | MAXRA     | # SET POSMAX IF OVFL.                     |
|          | BPL   |           | # CONTINUE WITH VALID RAPO.               |
|          |       | +3        |                                           |
| MAXRA    | DLOAD |           | # RAPO CALC IS NOT VALID. SET RAPO =      |
|          |       | NEARONE   | # POSMAX AS A TAG.                        |
| +3       | STORE | RAPO      | # APOGEE RADIUS M E:(-29) M:(-27)         |
| DUMPRPRA | RVQ   |           |                                           |
| #        |       |           | 30 W                                      |

```
1 # SUBROUTINE NAME: CALCTPER / CALCTFF DATE: 01.29.67
2 # MOD NO: 0 LOG SECTION: TIME OF FREE FALL
3
4 # MOD BY: RR BAIRNSFATHER
5 # MOD NO: 1 MOD BY: RR BAIRNSFATHER DATE: 21 MAR 67
6 # MOD NO: 2 MOD BY: RR BAIRNSFATHER DATE: 14 APR 67
7 # MOD BY: 3 MOD BY: RR BAIRNSFATHER DATE: 8 JUL 67 NEAR EARTH MUE AND NEG TFF (GONEPAST)
8 # MOD BY: 4 MOD BY: RR BAIRNSFATHER DATE: 21 NOV 67 ADD VARIABLE MU.
9 # MOD BY: 5 MOD BY: RR BAIRNSFATHER DATE: 21 MAR 68 ACCEPT DIFFERENT EARTH/MOON SCALES
10
11 # FUNCTIONAL DESCRIPTION: PROGRAM CALCULATES THE FREE-FALL TIME OF FLIGHT FROM PRESENT POSITION RN AND
12 # VELOCITY VN TO A RADIUS LENGTH SPECIFIED BY RTERM, SUPPLIED BY THE USER. THE POSITION VECTOR
13 # RN MAY BE ON EITHER SIDE OF THE CONIC, BUT RTERM IS CONSIDERED ON THE INBOUND SIDE.
14 # THE EQUATIONS ARE:
15 #
16 # Q2 = -SQRT(RTERM (2-RTERM ALFA) - LCP) (INBOUND SIDE) LEQ +- LCE/SQRT(ALFA)
17 #
18 # Q1 = $\bar{R}\bar{N} \cdot \bar{V}\bar{N} / \text{SQRT}(\text{MU})$ LEQ +- LCE/SQRT(ALFA)
19 #
20 # Z = NUM / DEN LEQ +- 1/SQRT(ALFA)
21 #
22 # WHERE, IF INBOUND
23 # NUM = RTERM -RN LEQ +- 2 LCE/ALFA
24 # DEN = Q2+Q1 LEQ +- 2 LCE/SQRT(ALFA)
25 #
26 # AND, IF OUTBOUND
27 # NUM = Q2-Q1 LEQ +- 2 LCE/SQRT(ALFA)
28 # DEN = 2 - ALFA (RTERM + RN). LEQ +- 2 LCE
29 #
30 # IF ALFA ZZ < 1.0 (FOR ALL CONICS EXCEPT ELLIPSES HAVING ABS(DEL ECC ANOM) G 90 DEG)
31 # THEN X = ALFA Z Z
32 # AND TFF = (RTERM +RN -2 ZZ T(X)) Z/SQRT(MU)
33 # EXCEPT IF ALFA PNZ, AND IF TFF NEG,
34 # THEN TFF = 2 PI /(ALFA SQRT(ALFA)) + TFF
35 # OR IF ALFA ZZ GEQ 1.0 (FOR ELLIPSES HAVING ABS(DEL ECC ANOM) GEQ 90 DEG)
36 # THEN X = 1/ALFA Z Z
37 # AND TFF = (PI/SQRT(ALFA) -Q2 +Q1 +2(X T(X) -1) /ALFA Z) /ALFA SQRT(MU)
38 # WHERE T(X) IS A POLYNOMIAL APPROXIMATION TO THE SERIES
39 #
40 # $1/3 - X/5 + X^2/7 - X^3/8 \dots$ (X < 1.0)
41 #
42 # CALLING SEQUENC: TIME TO RTERM TIME TO PERIGEE
43 # CALL CALL
44 # CALCTFF CALCTPER
45 # C(MPAC) = TERMNL RAD M C(MPAC) = PERIGEE RAD M
46 # FOR EITHER, E:(-29) M:(-27)
47 # FOR EITHER, PUSHLOC = PDL+0, ARBITRARY IF LEQ 8D.
```

```
1 #
2 #
3 # SUBROUTINES CALLED: T(X), VIA RTB
4 #
5 # NORMAL EXIT MODE: RVQ
6 # HOWEVER, PROGRAM EXITS WITH ONE OF THE FOLLOWING VALUES FOR TFF (-28) CS IN MPAC. USER MUST STORE.
7 # A. TFF = FLIGHT TIME. NORMAL CASE FOR POSITIVE FLIGHT TIME LESS THAN ONE ORBITAL PERIOD.
8 # B. (THIS OPTION IS NO LONGER USED.)
9 # C. TFF = POSMAX. THIS INDICATES THAT THE CONIC FROM THE PRESENT POSITION WILL NOT RETURN TO
10 # THE SPECIFIED ALTITUDE. ALSO INDICATES OUTBOUND PARABOLA OR HYPERBOLA.
11 #
12 # OUTPUT: C(MPAC) (-28) CS TIME OF FLIGHT, OR TIME TO PERIGEE
13 # TFFX (0) X LEFT FOR ENTRY DISPLAY TFF ROUTINES
14 # NRTERM E:(-29+NR) M RTERM, WEIGHTED BY NR LEFT FOR ENTRY DISPLAY TFF ROUTINES
15 # M:(-27+NR)
16 # TFFTEM E:(-59+2NR) LCP Z Z SGN(SDELF) LEFT FOR ENTRY DISPLAY TFF ROUTINES
17 # M:(-55+2NR) LCP /ALFA SGN(SDELF) LEFT FOR ENTRY DISPLAY TFF ROUTINES
18 # NOTE: TFFTEM = PDL 36D AND WILL BE DESTROYED BY .:UNIT:..
19 # RMAG1 E:(-29) M:(-27) PDL 12 NOT TOUCHED.
20 # TFFQ1 E:(-16) M:(-15) PDL 14D
21 # TFFDELQ E:(-16) M:(-15) PDL 10D
22 # PUSHLOC AT PDL+0
23 #
24 # ERASABLE INITIALIZATION REQUIRED:
25 # RONE E:(-29) M:(-27) M STATE VECTOR LEFT BY USER
26 # VONE' E:(+10) M:(+9) VN/SQRT(NU) LEFT BY TFF/CONIC
27 # RMAG1 E:(-29) M:(-27) PRESENT RADIUS, M LEFT BY TFFCONIC
28 # C(MPAC) E:(-29) M:(-27) RTERM, TERMINAL RADIUS LENGTH, M LEFT BY USER
29 #
30 # THE FOLLOWING ARE STORED IN THE PUSH LIST AREA.
31 # TFF/RTMU E:(17) M:(14) 1/SQRT(MU) LEFT BY TFFCONIC.
32 # NRMAG E:(-29+NR) M RMAG, NORMALIZED LEFT BY TFFCONIC
33 # M:(-27+NR)
34 # X1 -NR, NORM COUNT LEFT BY TFFCONIC
35 # TFFNP E:(-38+2NR) M LCP, SEMI LATUS RECTUM, WEIGHT NR LEFT BY TFFCONIC
36 # M:(-36+2N4)
37 # TFFALFA E:(26-NR) 1/M ALFA, WEIGHT NR LEFT BY TFFCONIC
38 # M:(24-NR)
39 # TFFRTALF E:(10+NA) SQRT(ALFA), NORMALIZED LEFT BY TFFCONIC
40 # M:(9+NA)
41 # X2 -NA, NORMCOUNT LEFT BY TFFCONIC
42 # TFF1/ALF E:(-22-2NA) SIGNED SEMI-MAJOR AXIS, WEIGHTED BY NA LEFT BY TFFCONIC
43 # M:(-20-2NA)
44 #
45 # DEBRIS: QPRET, PDL+0 ... PDL+3
46 # RTERM E:(-29) M:(-27) RTERM, TERMINAL RADIUS LENGTH
47 # RAPO E:(-29) M:(-27) PDL 16D (=NRTERM)
48 # RPER E:(-29) M:(-27) PDL 14D (=TFFQ1)
```

```
1 CALCTPER SETGO # ENTER WITH RPER IN MPAC
2
3 TFFSW
4 +3
5 CALCTFF CLEAR # ENTER WITH RTERM IN MPAC
6
7 TFFSW
8 +3 STORE # E:(-29) M:(-27)
9 SL*
10
11 0,1 # X1=-NR
12
13 NRTERM # RTERM E:(-29+NR) M:(-27+NR)
14 DMP BDSU
15 TFFALFA # ALFA E:(26-NR) M:(24-NR)
16
17 TFF1/4
18 DMP # (2-ALFA RTERM) (-3) TO PDL+0
19 NRTERM # E:(-29+NR) M:(-27+NR)
20
21 PDDL SR* # RTERM(2-ALFA RTERM) TO PDL+2
22
23 TFFNP # E:(-32+NR) M:(-30+NR)
24 0 # LC P E:(-38+2NR) M:(-36+2NR)
25 -6,1
26 DCOMP DAD # X1 = -NR
27
28 # DUE TO SHIFTS, KEEP PRECISION FOR SQRT
29
30 # RTERM(2-ALFA RTERM) FROM PDL +2
31
32 # E:(-32+NR) M:(-30+NR)
33 SR* # LEAVE E:(-32) M:(-30)
34
35 0,1 # X1 = -NR
36
37 BOFF DLOAD # CHECK TFF /TPER SWITCH
38 TFFSW
39 +2 # IF TFF, CONTINUE
40
41 TFFZEROS # IF TPER, SET Q2 = 0
42 SQRT # E:(-16) M:(-15)
43
44
45 MAXTFF1 # NO FREE FALL CONIC TO RTERM FROM HERE
46
47 # RESET PDL, SET TFF=POSMAX, AND EXIT.
48
49
50 DCOMP BOVB # RT IS ON INBOUND SIDE. ASSURE OVFIN=0
51 TCDANZIG # ANY PORT IN A STORM.
52
53 STOVL TFFTEM # Q2 E:(-16) M:(-15)
54
55 VONE' # VN/SQRT(MU) E:(10) M:(9)
56 SL3
57 RONE # SAVED RN. E:(-29) M:(-27)
58
59 STORE TFFQ1 # Q1, SAVE FOR GONEPAST TEST.
60
61 # E:(-16) M:(-15)
62
63 BMN BDSU
64
65 INBOUND # USE ALTERNATE Z
66 TFFTEM # Q2 E:(-16) M:(-15)
67
68
69 # OUTBOUND Z CALC CONTINUES HERE
70
71
72 STODL TFFX # NUM=Q2-Q1 E:(-16) M:(-15)
73
74 TFFALFA # ALFA E:(26-NR) M:(24-NR)
75
76 DMP BDSU
77
78
79
80
```

```
1
2 NRMAG # RMAG E:(-29+NR) M:(-27+NR)
3 # (2-RTERM ALFA) (-3) FROM PDL+0
4 SAVEDEN PUSH ABS # DEN TO PDL+0 E:(-3) OR (-16)
5 # M:(-3) OR (-15)
6 DAD BOV # INDETERMINANCY TEST
7 LIM(-22) # =1.0-B(-22)
8 TFFXTEST # GO IF DEN >= B(-22)
9 DLOAD PDDL # SET DEN=0 OTHERWISE
10 TFFZEROS
11 # XCH ZERO WITH PDL+0
12 DLOAD DCOMP
13 TFFALFA # ALFA E:(26-NR) M:(24-NR)
14 BMN DLOAD # FOR TPER: Z INDET AT DELE/2=0 AND 90.
15 TFFELL # ASSUME 90, AND LEAVE 0 IN PDL: 1/Z=D/N
16
17 # Z INDET. AT PERIGEE FOR PARAB OR HYPERB.
18 DUMPTFF1 RVQ # RETURN TFF =0
19
20 # INBOUND Z CALC CONTINUES HERE
21
22 INBOUND DLOAD # RESET PDL+0
23 DLOAD DSU # ALTERNATE Z CALC
24 RTERM # E:(-29) M:(-27)
25 RMAG1 # E:(-29) M:(-27)
26 STODL TFFX # NUM=RTERM-RN E:(-29) M:(-27)
27 TFFTEM # Q2 E:(-16) M:(-15)
28 DAD GOTO
29 TFFQ1 # Q1 E:(-16) M:(-15)
30 SAVEDEN # DEN = Q2+Q1 E:(-16) M:(-15)
31
32 TFFXTEST DAD PDDL # (ABS(DEN) TO PDL+2) E:(-3) OR (-16)
33 # M:(-3) OR (-15)
34 DP(-22) # RESTORE ABS(DEN) TO MPAC
35 TFFX # NUM E:(-16) OR (-29) M:(-15) OR (-27)
36 DMP SR*
37 TFFRTALF # SQRT(ALFA) E:(10+NA) M:(9+NA)
38 0 -3,2 # X2=-NA
39 DDV # C(MPAC) =NUM SQRT(ALFA) E:(-3) OR (-16)
40 # M:(-3) OR (-15)
41 # ABS(DEN) FROM PDL+2 E:(-3) OR (-16)
42 # M:(-3) OR (-15)
43 DLOAD BOV # (THE DLOAD IS SHARED WITH TFFELL)
44 TFFX # NUM E:(-16) OR (-29) M:(-15) OR (-27)
45 TFFELL # USE EQN FOR DELE GEQ 90, LEQ -90
46
47 # OTHERWISE, CONTINUE FOR GENERAL CONIC FOR TFF EQN
48
49 DDV STADR
50 # DEN FROM PDL+0 E:(-3) OR (-16)
51 # M:(-3) OR (-15)
52 STORE TFFTEM # Z SAVE FOR SIGN OF SDELf.
53
54
55
56
57
58
59
60
```



|    |          |       |          |                                           |
|----|----------|-------|----------|-------------------------------------------|
| 1  |          |       |          | # E:(-13) M:(-12)                         |
| 2  |          |       |          | # Z TO PDL+0                              |
| 3  |          | PUSH  | DSQ      |                                           |
| 4  |          | PUSH  | DMP      | # Z SQ TO PDL+2 E:(-26) M:(-24)           |
| 5  |          |       | TFFNP    | # LC P E:(-38+2NR) M:(-36+NR)             |
| 6  |          | SL    | SIGN     |                                           |
| 7  |          |       | 5        |                                           |
| 8  |          |       | TFFTEM   | # AFFIX SIGN FOR SDELF (ENTRY DISPLAY)    |
| 9  |          | STODL | TFFTEM   | # P ZSQ E:(-59+2NR) M:(-55+2NR)           |
| 10 |          |       |          | # (ARG IS USED IN TFF/TRIG)               |
| 11 |          |       |          | # ZSQ FROM PDL+2 E:(-26) M:(-24)          |
| 12 |          | PUSH  | DMP      | # RESTORE PUSH LOC                        |
| 13 |          |       | TFFALFA  | # ALFA E:(26-NR) M:(24-NR)                |
| 14 |          | SL*   |          |                                           |
| 15 |          |       | 0,1      | # X1=-NR                                  |
| 16 |          | STORE | TFFX     | # X                                       |
| 17 |          | RTB   | DMP      |                                           |
| 18 |          |       | T(X)     | # POLY                                    |
| 19 |          |       |          | # ZSQ FROM PDL+2 E:(-26) M:(-24)          |
| 20 |          | SR2   | BDSU     | # 2 ZSQ T(X) E:(-29) M:(-27)              |
| 21 |          |       | RTERM    | # RTERM E:(-29) M:(-27)                   |
| 22 |          | DAD   | DMP      |                                           |
| 23 |          |       | RMAG1    | # E:(-29) M:(-27)                         |
| 24 |          |       |          | # Z FROM PDL+0 E:(-13) M:(-12)            |
| 25 |          | SR3   | BPL      | # TFF SQRT(MU) E:(-45) M:(-42)            |
| 26 |          |       | ENDTFF   | # (NO PUSH UP)                            |
| 27 |          | PUSH  | SIGN     | # TFF SQRT(MU) TO PDL+0                   |
| 28 |          |       | TFFQ1    | # Q1 FOR GONEPAST TEST                    |
| 29 |          | BPL   | DLOAD    | # GONE PAST ?                             |
| 30 |          |       | NEGTF    | # YES. TFF < 0.                           |
| 31 |          |       | TFF1/ALF | # 1/ALFA E:(-22-2NA) M:(-20-2NA)          |
| 32 |          | DCOMP | BPL      | # ALFA > 0 ?                              |
| 33 |          |       | NEGTF    | # NO. TFF IS NEGATIVE.                    |
| 34 |          |       |          |                                           |
| 35 |          |       |          | # CORRECT FOR ORBITAL PERIOD.             |
| 36 |          |       |          |                                           |
| 37 |          | DCOMP |          | # YES. CORRECT FOR ORB PERIOD.            |
| 38 |          | DMP   | DDV      |                                           |
| 39 |          |       | PI/16    | # 2 PI (-5)                               |
| 40 |          |       | TFFRTALF | # SQRT(ALFA) E:(10+NA) M:(9+NA)           |
| 41 |          | SL*   | SL*      |                                           |
| 42 |          |       | 0 -4,2   | # X2=-NA                                  |
| 43 |          |       | 0 -4,2   |                                           |
| 44 |          | SL*   | DAD      |                                           |
| 45 |          |       | 0,2      |                                           |
| 46 |          |       |          | # TFF SQRT(MU) FROM PDL+0 E:(-45) M:(-42) |
| 47 | ENDTFF   | DMP   | BOV      | # TFF SQRT(MU) IN MPAC E:(-45) M:(-42)    |
| 48 |          |       | TFF/RTMU | # E:(17) M:(14)                           |
| 49 |          |       | MAXTFF   | # SET POSMAX IN OVFL.                     |
| 50 |          |       |          |                                           |
| 51 | DUMPTFF2 | RVQ   |          | # RETURN TFF (-28) CS IN MPAC.            |
| 52 |          |       |          |                                           |
| 53 |          |       |          |                                           |
| 54 |          |       |          |                                           |
| 55 |          |       |          |                                           |
| 56 |          |       |          |                                           |
| 57 |          |       |          |                                           |
| 58 |          |       |          |                                           |
| 59 |          |       |          |                                           |
| 60 |          |       |          |                                           |



```
1 NEGTF DLOAD
2
3 # TFF SQRT(MU) FROM PDL+0, NEGATIVE.
4
5 GOTO ENDTFF
6
7 MAXTFF1 DLOAD # RESET PDL
8 MAXTFF DLOAD RVQ
9 NEARONE
10
11 # TIME OF FLIGHT ELLIPSE WHEN DEL (ECCENTRIC ANOM) GEQ 90 AND LEQ -90.
12
13 # NUM FROM TFFX. E:(-16) OR (-29)
14 # M:(-15) OR (-27)
15 TFFELL SL2 # NUM E:(-14) OR (-27) M:(-13) OR (-25)
16 BDDV PUSH # TEMP SAVE D/N IN PDL+0
17 # DEN FROM PDL+0 E:(-3)/(-16) M:(-3)/(-15)
18 # N/D TO PDL+0 E:(11) M:(10)
19 TFFEL1 DLOAD DSU # (ENTER WITH D/N=0 IN PDL+0)
20 TFFTEM # Q2 E:(-16) M:(-15)
21 TFFQ1 # Q1 E:(-16) M:(-15)
22 STODL TFFDELQ # Q2-Q1 E:(-16) M:(-15)
23 # D/N FROM PDL+0
24 STADR
25 STORE TFFTEM # D/N E:(11) M:(10)
26 DMP SL*
27 TFF1/ALF # 1/ALFA E:(-22-2NA) M:(-20-2NA)
28 0,2 # 1/ALFA Z E:(-11-NA) M:(-10-NA)
29 PUSH DMP # TO PDL+0
30 TFFTEM # 1/Z E:(11) M:(10)
31 SL* BOVB
32 0,2 # X2= -NA
33 SIGNMPAC # IN CASE X= 1.0, CONTINUE
34 STORE TFFX # X=1/ALFA ZSQ
35 RTB DMP
36 T(X) # POLY
37 TFFX
38 SR3 DSU
39 DP2(-3)
40 DMP PUSH # 2(X T(X)-1) /Z ALFA E:(-15-NA)
41 # M:(-14-NA)
42 # 1/ALFA Z FROM PDL+0 E:(-11-NA)
43 # M:(-10-NA)
44 DLOAD DMP # GET SIGN FOR SDELF
45 TFFTEM # 1/Z E:(11) M:(10)
46 RMAG1 # E:(-29) M:(-27)
47 SL2 DAD
48 TFFQ1 # Q1 E:(-16) M:(-15)
49 STODL TFFTEM # (Q1+R 1/Z) =SGN OF SDELF E:(-16) M:(-15)
50 TFFNP # LC P E:(-38+2NR) M:(-36+2NR)
51 DMP SL* # CALC FOR ARG FOR TFF/TRIG.
```

|    | THE | 12 | 14 |
|----|-----|----|----|
| 1  |     |    |    |
| 2  |     |    |    |
| 3  |     |    |    |
| 4  |     |    |    |
| 5  |     |    |    |
| 6  |     |    |    |
| 7  |     |    |    |
| 8  |     |    |    |
| 9  |     |    |    |
| 10 |     |    |    |
| 11 |     |    |    |
| 12 |     |    |    |
| 13 |     |    |    |
| 14 |     |    |    |
| 15 |     |    |    |
| 16 |     |    |    |
| 17 |     |    |    |
| 18 |     |    |    |
| 19 |     |    |    |
| 20 |     |    |    |
| 21 |     |    |    |
| 22 |     |    |    |
| 23 |     |    |    |
| 24 |     |    |    |
| 25 |     |    |    |
| 26 |     |    |    |
| 27 |     |    |    |
| 28 |     |    |    |
| 29 |     |    |    |
| 30 |     |    |    |
| 31 |     |    |    |
| 32 |     |    |    |
| 33 |     |    |    |
| 34 |     |    |    |
| 35 |     |    |    |
| 36 |     |    |    |
| 37 |     |    |    |
| 38 |     |    |    |
| 39 |     |    |    |
| 40 |     |    |    |
| 41 |     |    |    |
| 42 |     |    |    |
| 43 |     |    |    |
| 44 |     |    |    |
| 45 |     |    |    |
| 46 |     |    |    |
| 47 |     |    |    |
| 48 |     |    |    |
| 49 |     |    |    |
| 50 |     |    |    |
| 51 |     |    |    |
| 52 |     |    |    |
| 53 |     |    |    |
| 54 |     |    |    |
| 55 |     |    |    |
| 56 |     |    |    |
| 57 |     |    |    |
| 58 |     |    |    |
| 59 |     |    |    |
| 60 |     |    |    |
| 61 |     |    |    |
| 62 |     |    |    |
| 63 |     |    |    |
| 64 |     |    |    |
| 65 |     |    |    |
| 66 |     |    |    |
| 67 |     |    |    |
| 68 |     |    |    |
| 69 |     |    |    |
| 70 |     |    |    |
| 71 |     |    |    |
| 72 |     |    |    |
| 73 |     |    |    |
| 74 |     |    |    |
| 75 |     |    |    |
| 76 |     |    |    |
| 77 |     |    |    |
| 78 |     |    |    |
| 79 |     |    |    |
| 80 |     |    |    |

```
1 # PROGRAM NAME: T(X) DATE: 01.17.67
2 # MOD NO: 0 LOG SECTION: TIME OF FREE FALL
3
4 # MOD BY: RR BAIRNSFATHER
5 #
6 # FUNCTIONAL DESCRIPTION: THE POLYNOMIAL T(X) IS USED BY TIME OF FLIGHT SUBROUTINES CALCTFF AND
7 # CALCTPER TO APPROXIMATE THE SERIES
8 # 2 3
9 # 1/3 -X/5 +X /7 -X /9 ...
10 #
11 # WHERE X = ALFA Z Z IF ALFA Z Z LEQ 1
12 # X = 1/(ALFA Z Z) IF ALFA Z Z G 1
13 #
14 # ALSO X IS NEG FOR HYPERBOLIC ORBITS
15 # X = 0 FOR PARABOLIC ORBITS
16 # X IS POSITIVE FOR ELLIPTIC ORBITS
17 #
18 # FOR FLIGHT 278, THE POLYNOMIAL T(X) IS FITTED OVER THE RANGE (0,+1) AND HAS A MAXIMUM
19 # DEVIATION FROM THE SERIES OF 2 E-5. (T(X) IS A CHEBYCHEV TYPE FIT AND WAS OBTAINED USING
20 # MAX PROGRAM AUTOCURFIT294RRB AND IS VALID TO THE SAME TOLERANCE OVER THE RANGE (-.08,+1).)
21 #
22 # CALLING SEQUENCE: RTB
23 # T(X)
24 # C(MPAC) = X
25 #
26 # SUBROUTINE CALLED: NONE
27 #
28 # NORMAL EXIT MODE: TC TANZIG
29 #
30 # ALARMS: NONE
31 #
32 # OUTPUT: C(MPAC) = T(X)
33 #
34 # ERASABLE INITIALIZATION REQUIRED:
35 # C(MPAC) = X
36 #
37 # DEBRIS: NONE
38
39 T(X) TC POLY
40 DEC 4 # N-1
41 2DEC 3.333333333 E-1
42
43 2DEC* -1.999819135 E-1*
44
45 2DEC* 1.418148467 E-1*
46
47 2DEC* -1.01310997 E-1*
48
49 2DEC* 5.609004986 E-2*
50
51 2DEC* -1.536156925 E-2*
52
53 ENDT(X) TC DANZIG
54
55 TCDANZIG = ENDT(X)
```

# TIME\_OF\_FREE\_FALL

# TFF CONSTANTS

BANK 32

SETLOC TOF-FF1

BANK

# # NOTE: ADJUSTED MUE FOR NEAR EARTH TRAJ.

#MUE = 3.990815471 E10 # M CUBE/CS SQ

#RTMUE = 1.997702549 E5 B-18\* # MODIFIED EARTH MU

#

# # NOTE: ADJUSTED MUE FOR NEAR EARTH TRAJ.

#MUM = 4.902778 E8 # M CUBE/CS SQ

#RTMUM 2DEC\* 2.21422176 E4 B-18\*

PI/16 2DEC 3.141592653 B-4

LIM(-22) 2OCT 3777737700 # 1.0 -B(-22)

DP(-22) 2OCT 0000000100 # B(-22)

DP2(-3) 2DEC 1 B-3

DP2(-4) 2DEC 1 B-4 # 1/16

# RPAD1 2DEC 6373338 B-29 # M (-29) = 20909901.57 FT

RPAD1 = RPAD

R300K 2DEC 6464778 B-29 # (-29) M

NEARONE 2DEC .999999999

TFFZEROS EQUALS HI6ZEROS

TFF1/4 EQUALS HIDP1/4

# PROGRAM DESCRIPTION  
# PROGRAM NAME -- SELF-CHECK

DATE: 20 DECEMBER 1967  
LOG SECTION: AGC BLOCK TWO SELF-CHECK  
ASSEMBLY SUBROUTINE UTILITYM REV 25

# MOD NO -- 1  
# MOD BY -- GAUNTT  
#

## # FUNCTIONAL DESCRIPTION

# PROGRAM HAS TWO MAIN PARTS. THE FIRST IS SELF-CHECK WHICH RUNS AS A ZERO PRIORITY JOB WITH NO CORE SET, AS  
# PART OF THE BACK-UP IDLE LOOP. THE SECOND IS SHOW-BANKSUM WHICH RUNS AS A REGULAR EXECUTIVE JOB WITH ITS OWN  
# STARTING VERB.  
# THE PURPOSE OF SELF-CHECK IS TO CHECK OUT VARIOUS PARTS OF THE COMPUTER AS OUTLINED BELOW IN THE OPTIONS.  
# THE PURPOSE OF SHOW-BANKSUM IS TO DISPLAY THE SUM OF EACH BANK, ONE AT A TIME.  
# IN ALL THERE ARE 7 POSSIBLE OPTIONS IN THIS BLOCK II VERSION OF SELF-CHECK. MORE DETAIL DESCRIPTION MAY BE  
# FOUND IN E-2065 BLOCK II AGC SELF-CHECK AND SHOW BANKSUM BY EDWIN D. SMALLY DECEMBER 1966, AND ADDENDA 2 AND 3.  
# THE DIFFERENT OPTIONS ARE CONTROLLED BY PUTTING DIFFERENT NUMBERS IN THE SMODE REGISTER (NOUN 27). BELOW IS  
# A DESCRIPTION OF WHAT PARTS OF THE COMPUTER THAT ARE CHECKED BY THE OPTIONS, AND THE CORRESPONDING NUMBER, IN  
# OCTAL, TO LOAD INTO SMODE.  
# +-4 ERASABLE MEMORY  
# +-5 FIXED MEMORY  
# +-1,2,3,6,7,10 EVERYTHING IN OPTIONS 4 AND 5.  
# -0 SAME AS +-10 UNTIL AN ERROR IS DETECTED.  
# +0 NO CHECK, PUTS COMPUTER INTO THE BACKUP IDLE LOOP.

## # WARNINGS

# USE OF E MEMORY RESERVED FOR SELF-CHECK (EVEN IN IDLE LOOP) AS TEMP STORAGE BY OTHER PROGRAMS IS DANGEROUS.  
# SMODE SET GREATER THAN OCT 10 PUTS COMPUTER INTO BACKUP IDLE LOOP.

## # CALLING SEQUENCE

# TO CALL SELF-CHECK KEY IN  
# V 21 N 27 E OPTION NUMBER E  
# TO CALL SHOW-BANKSUM KEY IN  
# V 91 E DISPLAYS FIRST BANK  
# V 33 E PROCEED, DISPLAYS NEXT BANK

## # EXIT MODES, NORMAL AND ALARM

# SELF-CHECK NORMALLY CONTINUES INDEFINITELY UNLESS THERE IS AN ERROR DETECTED. IF SO + OPTION NUMBERS PUT  
# COMPUTER INTO BACKUP IDLE LOOP, - OPTION NUMBERS RESTART THE OPTION.

# THE -0 OPTION PROCEEDS FROM THE LINE FOLLOWING THE LINE WHERE THE ERROR WAS DETECTED.  
# SHOW-BANKSUM PROCEEDS UNTIL A TERMINATE IS KEYED IN (V 34 E). THE COMPUTER IS PUT INTO THE BACKUP IDLE LOOP.

## # OUTPUT

```
1 # SELF-CHECK UPON DETECTING AN ERROR LOADS THE SELF-CHECK ALARM CONSTANT (01102) INTO THE FAILREG SET AND
2 #
3 # TURNS ON THE ALARM LIGHT. THE OPERATOR MAY THEN DISPLAY THE THREE FAILREGS BY KEYING IN V 05 N 09 E. FOR FURTHER
4 # INFORMATION HE MAY KEY IN V 05 N 08 E, THE DSKY DISPLAY IN R1 WILL BE ADDRESS+1 OF WHERE THE ERROR WAS DETECTED,
5 # IN R2 THE BBCON OF SELF-CHECK, AND IN R3 THE TOTAL NUMBER OF ERRORS DETECTED BY SELF-CHECK SINCE THE LAST MAN
6 # INITIATED FRESH START (SLAP1).
7 # SHOW-BANKSUM STARTING WITH BANK 0 DISPLAYS IN R1 THE BANK SUM (A +-NUMBER EQUAL TO THE BANK NUMBER), IN R2
8 # THE BANK NUMBER, AND IN R3 THE BUGGER WORD.
9 #
10 #
11 # ERASABLE INITIALIZATION REQUIRED
12 #
13 # ACCOMPLISHED BY FRESH START
14 # SMODE SET TO +0
15 #
16 #
17 # DEBRIS
18 #
19 # ALL EXITS FROM THE CHECK OF ERASABLE (ERASCHK) RESTORE ORIGINAL CONTENTS TO REGISTERS UNDER CHECK.
20 # EXCEPTION IS A RESTART. RESTART THAT OCCURS DURING ERASCHK RESTORES ERASABLE, UNLESS THERE IS EVIDENCE TO DOUBT
21 # E MEMORY, IN WHICH CASE PROGRAM THEN DOES A FRESH START (DOFSTART).
```

```
22
23
24 BANK 25
25 SETLOC SELFCHC
26 BANK
```

```
27
28 COUNT* $$/SELF
```

```
29 SBIT1 EQUALS BIT1
30 SBIT2 EQUALS BIT2
31 SBIT3 EQUALS BIT3
32 SBIT4 EQUALS BIT4
33 SBIT5 EQUALS BIT5
34 SBIT6 EQUALS BIT6
35 SBIT7 EQUALS BIT7
36 SBIT8 EQUALS BIT8
37 SBIT9 EQUALS BIT9
38 SBIT10 EQUALS BIT10
39 SBIT11 EQUALS BIT11
40 SBIT12 EQUALS BIT12
41 SBIT13 EQUALS BIT13
42 SBIT14 EQUALS BIT14
43 SBIT15 EQUALS BIT15
44
45 S+ZERO EQUALS ZERO
46 S+1 EQUALS BIT1
47 S+2 EQUALS BIT2
48 S+3 EQUALS THREE
49 S+4 EQUALS FOUR
50 S+5 EQUALS FIVE
51 S+6 EQUALS SIX
```

|          |        |          |                                                                                                                                                               |
|----------|--------|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| S+7      | EQUALS | SEVEN    |                                                                                                                                                               |
| S8BITS   | EQUALS | LOW8     | # 00377                                                                                                                                                       |
| CNTRCON  | =      | OCT50    | # USED IN CNTRCHK                                                                                                                                             |
| ERASCON1 | OCTAL  | 00061    | # USED IN ERASCHK                                                                                                                                             |
| ERASCON2 | OCTAL  | 01373    | # USED IN ERASCHK                                                                                                                                             |
| ERASCON6 | =      | OCT1400  | # USED IN ERASCHK                                                                                                                                             |
| ERASCON3 | OCTAL  | 01461    | # USED IN ERASCHK                                                                                                                                             |
| ERASCON4 | OCTAL  | 01773    | # USED IN ERASCHK                                                                                                                                             |
| S10BITS  | EQUALS | LOW10    | # 01777, USED IN ERASCHK                                                                                                                                      |
| SBNK03   | EQUALS | PRI06    | # 06000, USED IN ROPECHK                                                                                                                                      |
| -MAXADRS | =      | HI5      | # FOR ROPECHK                                                                                                                                                 |
| SIXTY    | OCTAL  | 00060    |                                                                                                                                                               |
| SUPRCON  | OCTAL  | 60017    | # USED IN ROPECHK                                                                                                                                             |
| S13BITS  | OCTAL  | 17777    |                                                                                                                                                               |
| CONC+S1  | OCTAL  | 25252    | # USED IN CYCLSHFT                                                                                                                                            |
| CONC+S2  | OCTAL  | 52400    | # USED IN CYCLSHFT                                                                                                                                            |
| ERASCON5 | OCTAL  | 76777    |                                                                                                                                                               |
| S-7      | =      | OCT77770 |                                                                                                                                                               |
| S-4      | EQUALS | NEG4     |                                                                                                                                                               |
| S-3      | EQUALS | NEG3     |                                                                                                                                                               |
| S-2      | EQUALS | NEG2     |                                                                                                                                                               |
| S-1      | EQUALS | NEGONE   |                                                                                                                                                               |
| S-ZERO   | EQUALS | NEG0     |                                                                                                                                                               |
|          | EBANK= | LST1     |                                                                                                                                                               |
| ADRS1    | ADRES  | SKEEP1   |                                                                                                                                                               |
| SELFADRS | ADRES  | SELFCHK  | # SELFCHK RETURN ADDRESS. SHOULD BE PUT<br># IN SELFRET WHEN GOING FROM SELFCHK TO<br># SHOWSUM AND PUT IN SKEEP1 WHEN GOING<br># FROM SHOWSUM TO SELF-CHECK. |
| PRERRORS | CA     | ERESTORE | # IS IT NECESSARY TO RESTORE ERASABLE                                                                                                                         |
|          | EXTEND |          |                                                                                                                                                               |
|          | BZF    | ERRORS   | # NO                                                                                                                                                          |
|          | EXTEND |          |                                                                                                                                                               |
|          | DCA    | SKEEP5   |                                                                                                                                                               |
|          | INDEX  | SKEEP7   |                                                                                                                                                               |
|          | DXCH   | 0000     | # RESTORE THE TWO ERASABLE REGISTERS                                                                                                                          |
|          | CA     | S+ZERO   |                                                                                                                                                               |
|          | TS     | ERESTORE |                                                                                                                                                               |
| ERRORS   | INHINT |          |                                                                                                                                                               |
|          | CA     | Q        |                                                                                                                                                               |
|          | TS     | SFAIL    | # SAVE Q FOR FAILURE LOCATION                                                                                                                                 |
|          | TS     | ALMCADR  | # FOR DISPLAY WITH BBANK AND ERCOUNT                                                                                                                          |
|          | INCR   | ERCOUNT  | # KEEP TRACK OF NUMBER OF MALFUNCTIONS.                                                                                                                       |
| TCALARM2 | TC     | ALARM2   |                                                                                                                                                               |
|          | OCT    | 01102    | # SELF-CHECK MALFUNCTION INDICATOR                                                                                                                            |
| SIDLOOP  | CCS    | SMODE    |                                                                                                                                                               |
|          | CA     | S+ZERO   |                                                                                                                                                               |
|          | TS     | SMODE    |                                                                                                                                                               |

```
1
2 TC SELFCHK # GO TO IDLE LOOP
3 TC SFAIL # CONTINUE WITH SELF-CHECK
4
5 -1CHK CCS A
6 TCF PRERRORS
7
8 TCF PRERRORS
9 CCS A
10 TCF PRERRORS
11
12 TC Q
13
14 SMODECHK EXTEND
15 QXCH SKEEP1
16 TC CHECKNJ # CHECK FOR NEW JOB
17 CCS SMODE
18
19 TC SOPTIONS
20 TC SMODECHK +2 # TO BACKUP IDLE LOOP
21 TC SOPTIONS
22
23 INCR SCOUNT
24 TC SKEEP1 # CONTINUE WITH SELF-CHECK
25
26 SOPTIONS AD S-7
27 EXTEND
28 BZMF +2 # FOR OPTIONS BELOW NINE.
29
30 BNKOPTN TC SIDLOOP # ILLEGAL OPTION. GO TO IDLE LOOP.
31 INCR SCOUNT # FOR OPTIONS BELOW NINE.
32 AD S+7
33
34
35 INDEX A
36 TC SOPTION1
37
38 SOPTION1 TC SKEEP1 # WAS TC+TCF
39 SOPTION2 TC SKEEP1 # WAS IN:OUT1
40 SOPTION3 TC SKEEP1 # WAS COUNTCHK
41
42 SOPTION4 TC ERASCHK
43 SOPTION5 TC ROPECHK
44 SOPTION6 TC SKEEP1
45
46 SOPTION7 TC SKEEP1
47 SOPTON10 TC SKEEP1 # CONTINUE WITH SELF-CHECK
48
49 CHECKNJ EXTEND
50 QXCH SELFRET # SAVE RETURN ADDRESS WHILE TESTING NEWJOB
51 TC POSTJUMP # TO SEE IF ANY JOBS HAVE BECOME ACTIVE.
52 CADR ADVAN
53
54 SELFCHK TC SMODECHK # ** CHARLEY, COME IN HERE
55
56 # SKEEP7 HOLDS LOWEST OF TWO ADDRESSES BEING CHECKED.
57 # SKEEP6 HOLDS B(X+1).
58 # SKEEP5 HOLDS B(X).
59 # SKEEP4 HOLDS C(EBANK) DURING ERASLOOP AND CHECKNJ.
60 # SKEEP3 HOLDS LAST ADDRESS BEING CHECKED (HIGHEST ADDRESS).
```



# SKEEP2 CONTROLS CHECKING OF NON-SWITCHABLE ERASABLE MEMORY WITH BANK NUMBERS IN EB.  
# ERASCHK TAKES APPROXIMATELY 7 SECONDS

|         |    |          |                        |
|---------|----|----------|------------------------|
| ERASCHK | CA | S+1      |                        |
|         | TS | SKEEP2   |                        |
| OEBANK  | CA | S+ZERO   |                        |
|         | TS | EBANK    |                        |
|         | CA | ERASCON3 | # 01461                |
|         | TS | SKEEP7   | # STARTING ADDRESS     |
|         | CA | S10BITS  | # 01777                |
|         | TS | SKEEP3   | # LAST ADDRESS CHECKED |
|         | TC | ERASLOOP |                        |

|          |    |          |                        |
|----------|----|----------|------------------------|
| E134567B | CA | ERASCON6 | # 01400                |
|          | TS | SKEEP7   | # STARTING ADDRESS     |
|          | CA | S10BITS  | # 01777                |
|          | TS | SKEEP3   | # LAST ADDRESS CHECKED |
|          | TC | ERASLOOP |                        |

|        |    |          |                        |
|--------|----|----------|------------------------|
| 2EBANK | CA | ERASCON6 | # 01400                |
|        | TS | SKEEP7   | # STARTING ADDRESS     |
|        | CA | ERASCON4 | # 01773                |
|        | TS | SKEEP3   | # LAST ADDRESS CHECKED |
|        | TC | ERASLOOP |                        |

|         |    |          |                        |
|---------|----|----------|------------------------|
| NOEBANK | TS | SKEEP2   | # +0                   |
|         | CA | ERASCON1 | # 00061                |
|         | TS | SKEEP7   | # STARTING ADDRESS     |
|         | CA | ERASCON2 | # 01373                |
|         | TS | SKEEP3   | # LAST ADDRESS CHECKED |

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| ERASLOOP | INHINT |          |                                           |
|          | CA     | EBANK    | # STORES C(EBANK)                         |
|          | TS     | SKEEP4   |                                           |
|          | EXTEND |          |                                           |
|          | NDX    | SKEEP7   |                                           |
|          | DCA    | 0000     |                                           |
|          | DXCH   | SKEEP5   | # STORES C(X) AND C(X+1) IN SKEEP6 AND 5. |
|          | CA     | SKEEP7   |                                           |
|          | TS     | ERESTORE | # IF RESTART, RESTORE C(X) AND C(X+1)     |
|          | TS     | L        |                                           |
|          | INCR   | L        |                                           |
|          | NDX    | A        |                                           |
|          | DXCH   | 0000     | # PUTS OWN ADDRESS IN X AND X +1          |
|          | NDX    | SKEEP7   |                                           |
|          | CS     | 0001     | # CS X+1                                  |
|          | NDX    | SKEEP7   |                                           |
|          | AD     | 0000     | # AD X                                    |
|          | TC     | -1CHK    |                                           |
|          | CA     | ERESTORE | # HAS ERASABLE BEEN RESTORED              |
|          | EXTEND |          |                                           |

|    |                                                                            |        |          |                                           |
|----|----------------------------------------------------------------------------|--------|----------|-------------------------------------------|
| 1  |                                                                            |        |          |                                           |
| 2  |                                                                            | BZF    | ELOOPFIN | # YES, EXIT ERASLOOP.                     |
| 3  |                                                                            | EXTEND |          |                                           |
| 4  |                                                                            | NDX    | SKEEP7   |                                           |
| 5  |                                                                            | DCS    | 0000     | # COMPLEMENT OF ADDRESS OF X AND X+1      |
| 6  |                                                                            | NDX    | SKEEP7   |                                           |
| 7  |                                                                            | DXCH   | 0000     | # PUT COMPLEMENT OF ADDRESS OF X AND X+1  |
| 8  |                                                                            | NDX    | SKEEP7   |                                           |
| 9  |                                                                            | CS     | 0000     | # CS X                                    |
| 10 |                                                                            | NDX    | SKEEP7   |                                           |
| 11 |                                                                            | AD     | 0001     | # AD X+1                                  |
| 12 |                                                                            | TC     | -1CHK    |                                           |
| 13 |                                                                            | CA     | ERESTORE | # HAS ERASABLE BEEN RESTORED              |
| 14 |                                                                            | EXTEND |          |                                           |
| 15 |                                                                            | BZF    | ELOOPFIN | # YES, EXIT ERASLOOP.                     |
| 16 |                                                                            | EXTEND |          |                                           |
| 17 |                                                                            | DCA    | SKEEP5   |                                           |
| 18 |                                                                            | NDX    | SKEEP7   |                                           |
| 19 |                                                                            | DXCH   | 0000     | # PUT B(X) AND B(X+1) BACK INTO X AND X+1 |
| 20 |                                                                            | CA     | S+ZERO   |                                           |
| 21 |                                                                            | TS     | ERESTORE | # IF RESTART, DO NOT RESTORE C(X), C(X+1) |
| 22 | ELOOPFIN                                                                   | RELINT |          |                                           |
| 23 |                                                                            | TC     | CHECKNJ  | # CHECK FOR NEW JOB                       |
| 24 |                                                                            | CA     | SKEEP4   | # REPLACES B(EBANK)                       |
| 25 |                                                                            | TS     | EBANK    |                                           |
| 26 |                                                                            | INCR   | SKEEP7   |                                           |
| 27 |                                                                            | CS     | SKEEP7   |                                           |
| 28 |                                                                            | AD     | SKEEP3   |                                           |
| 29 |                                                                            | EXTEND |          |                                           |
| 30 |                                                                            | BZF    | +2       |                                           |
| 31 |                                                                            | TC     | ERASLOOP | # GO TO NEXT ADDRESS IN SAME BANK         |
| 32 |                                                                            | CCS    | SKEEP2   |                                           |
| 33 |                                                                            | TC     | NOEBANK  |                                           |
| 34 |                                                                            | INCR   | SKEEP2   | # PUT +1 IN SKEEP2.                       |
| 35 |                                                                            | CA     | EBANK    |                                           |
| 36 |                                                                            | AD     | SBIT9    |                                           |
| 37 |                                                                            | TS     | EBANK    |                                           |
| 38 |                                                                            | AD     | ERASCON5 | # 76777, CHECK FOR BANK E2                |
| 39 |                                                                            | EXTEND |          |                                           |
| 40 |                                                                            | BZF    | 2EBANK   |                                           |
| 41 |                                                                            | CCS    | EBANK    |                                           |
| 42 |                                                                            | TC     | E134567B | # GO TO EBANKS 1,3,4,5,6, AND 7           |
| 43 |                                                                            | CA     | ERASCON6 | # END OF ERASCHK                          |
| 44 |                                                                            | TS     | EBANK    |                                           |
| 45 | # CNTRCHK PERFORMS A CS OF ALL REGISTERS FROM OCT. 60 THROUGH OCT. 10.     |        |          |                                           |
| 46 | # INCLUDED ARE ALL COUNTERS, T6-1, CYCLE AND SHIFT, AND ALL RUPT REGISTERS |        |          |                                           |
| 47 | CNTRCHK                                                                    | CA     | CNTRCON  | # 00050                                   |
| 48 | CNTRLOOP                                                                   | TS     | SKEEP2   |                                           |
| 49 |                                                                            | AD     | SBIT4    | # +10 OCTAL                               |
| 50 |                                                                            | INDEX  | A        |                                           |
| 51 |                                                                            | CS     | 0000     |                                           |
| 52 |                                                                            |        |          |                                           |
| 53 |                                                                            |        |          |                                           |
| 54 |                                                                            |        |          |                                           |
| 55 |                                                                            |        |          |                                           |
| 56 |                                                                            |        |          |                                           |
| 57 |                                                                            |        |          |                                           |
| 58 |                                                                            |        |          |                                           |
| 59 |                                                                            |        |          |                                           |
| 60 |                                                                            |        |          |                                           |

```
1
2 CCS SKEEP2
3 TC CNTRLOOP
4
5 # CYCLSHFT CHECKS THE CYCLE AND SHIFT REGISTERS
6 CYCLSHFT CA CONC+S1 # 25252
7 TS CYR # C(CYR) = 12525
8 TS CYL # C(CYL) = 52524
9 TS SR # C(SR) = 12525
10 TS EDOP # C(EDOP) = 00125
11 AD CYR # 37777 C(CYR) = 45252
12 AD CYL # 00-12524 C(CYL) = 25251
13 AD SR # 00-25251 C(SR) = 05252
14 AD EDOP # 00-25376 C(EDOP) = +0
15 AD CONC+S2 # C(CONC+S2) = 52400
16 TC -1CHK
17 AD CYR # 45252
18 AD CYL # 72523
19 AD SR # 77775
20 AD EDOP # 77775
21 AD S+1 # 77776
22 TC -1CHK
23
24 INCR SCOUNT +1
25 TC SMODECHK
26
27 # SKEEP1 HOLDS SUM
28 # SKEEP2 HOLDS PRESENT CONTENTS OF ADDRESS IN ROPECHK AND SHOWSUM ROUTINES
29 # SKEEP2 HOLDS BANK NUMBER IN LOW ORDER BITS DURING SHOWSUM DISPLAY
30 # SKEEP3 HOLDS PRESENT ADDRESS (00000 TO 01777 IN COMMON FIXED BANKS)
31 # (04000 TO 07777 IN FXFX BANKS)
32 # SKEEP3 HOLDS BUGGER WORD DURING SHOWSUM DISPLAY
33 # SKEEP4 HOLDS BANK NUMBER AND SUPER BANK NUMBER
34 # SKEEP5 COUNTS 2 SUCCESSIVE TC SELF WORDS
35 # SKEEP6 CONTROLS ROPECHK OR SHOWSUM OPTION
36 # SKEEP7 CONTROLS WHEN ROUTINE IS IN COMMON FIXED OR FIXED FIXED BANKS
37
38 ROPECHK CA S-ZERO # *
39 TS SKEEP6 # * -0 FOR ROPECHK.
40 STSHOSUM CA S+ZERO # * SHOULD BE ROPECHK
41
42 TS SKEEP4 # BANK NUMBER
43
44 COMMFX CA S+1
45 TS SKEEP7
46 CA S+ZERO
47 TS SKEEP1
48 TS SKEEP3
49 CA S+1
50 COMADRS TS SKEEP5 # COUNTS DOWN 2 TC SELF WORDS
51 CA SKEEP4
52 TS L # TO SET SUPER BANK
53 MASK HI5
54
55
56
57
58
59
60
```

|         |        |            |                                         |
|---------|--------|------------|-----------------------------------------|
|         | AD     | SKEEP3     |                                         |
|         | TC     | SUPDACAL   | # SUPER DATA CALL                       |
|         | TC     | ADSUM      |                                         |
|         | AD     | SBIT11     | # 02000                                 |
|         | TC     | ADRSCHK    |                                         |
| FXFX    | CS     | A          |                                         |
|         | TS     | SKEEP7     |                                         |
|         | EXTEND |            |                                         |
|         | BZF    | +3         |                                         |
|         | CA     | SBIT12     | # 04000, STARTING ADDRESS OF BANK 02    |
|         | TC     | +2         |                                         |
|         | CA     | SBNK03     | # 06000, STARTING ADDRESS OF BANK 03    |
|         | TS     | SKEEP3     |                                         |
|         | CA     | S+ZERO     |                                         |
|         | TS     | SKEEP1     |                                         |
|         | CA     | S+1        |                                         |
|         | TS     | SKEEP5     | # COUNTS DOWN 2 TC SELF WORDS           |
| FXADRS  | INDEX  | SKEEP3     |                                         |
|         | CA     | 0000       |                                         |
|         | TC     | ADSUM      |                                         |
|         | TC     | ADRSCHK    |                                         |
| ADSUM   | TS     | SKEEP2     |                                         |
|         | AD     | SKEEP1     |                                         |
|         | TS     | SKEEP1     |                                         |
|         | CAF    | S+ZERO     |                                         |
|         | AD     | SKEEP1     |                                         |
|         | TS     | SKEEP1     |                                         |
|         | CS     | SKEEP2     |                                         |
|         | AD     | SKEEP3     |                                         |
|         | TC     | Q          |                                         |
| ADRSCHK | LXCH   | A          |                                         |
|         | CA     | SKEEP3     |                                         |
|         | MASK   | LOW10      | # RELATIVE ADDRESS                      |
|         | AD     | -MAXADRS   | # SUBTRACT MAX RELATIVE ADDRESS = 1777. |
|         | EXTEND |            |                                         |
|         | BZF    | SOPTION    | # CHECKSUM FINISHED IF LAST ADDRESS.    |
|         | CCS    | SKEEP5     | # IS CHECKSUM FINISHED                  |
|         | TC     | +3         | # NO                                    |
|         | TC     | +2         | # NO                                    |
|         | TC     | SOPTION    | # GO TO ROPECHK SHOWSUM OPTION          |
|         | CCS    | L          | # -0 MEANS A TC SELF WORD.              |
|         | TC     | CONTINU    |                                         |
|         | TC     | CONTINU    |                                         |
|         | TC     | CONTINU    |                                         |
|         | CCS    | SKEEP5     |                                         |
|         | TC     | CONTINU +1 |                                         |
|         | CA     | S-1        |                                         |

|          |        |            |                                            |
|----------|--------|------------|--------------------------------------------|
| CONTINU  | TC     | CONTINU +1 | # AD IN THE BUGGER WORD                    |
|          | CA     | S+1        | # MAKE SURE TWO CONSECUTIVE TC SELF WORDS  |
|          | TS     | SKEEP5     |                                            |
|          | CCS    | SKEEP6     | # *                                        |
|          | CCS    | NEWJOB     | # * +1, SHOWSUM                            |
|          | TC     | CHANG1     | # *                                        |
|          | TC     | +2         | # *                                        |
|          | TC     | CHECKNJ    | # -0 IN SKEEP6 FOR ROPECHK                 |
| ADRS+1   | INCR   | SKEEP3     |                                            |
|          | CCS    | SKEEP7     |                                            |
|          | TC     | COMADRS    |                                            |
|          | TC     | COMADRS    |                                            |
|          | TC     | FXADRS     |                                            |
|          | TC     | FXADRS     |                                            |
| NXTBNK   | CS     | SKEEP4     |                                            |
|          | AD     | LSTBNKCH   | # LAST BANK TO BE CHECKED                  |
|          | EXTEND |            |                                            |
|          | BZF    | ENDSUMS    | # END OF SUMMING OF BANKS.                 |
|          | CA     | SKEEP4     |                                            |
|          | AD     | SBIT11     |                                            |
|          | TS     | SKEEP4     | # 37 TO 40 INCRMTS SKEEP4 BY END RND CARRY |
| 17TO20   | TC     | CHKSUPR    |                                            |
|          | CA     | SBIT15     |                                            |
|          | ADS    | SKEEP4     | # SET FOR BANK 20                          |
| CHKSUPR  | TC     | GONXTBNK   |                                            |
|          | MASK   | HI5        |                                            |
|          | EXTEND |            |                                            |
| 27TO30   | BZF    | NXTSUPR    | # INCREMENT SUPER BANK                     |
|          | AD     | S13BITS    |                                            |
|          | EXTEND |            |                                            |
|          | BZF    | +2         | # BANK SET FOR 30                          |
|          | TC     | GONXTBNK   |                                            |
|          | CA     | SIXTY      | # FIRST SUPER BANK                         |
|          | ADS    | SKEEP4     |                                            |
|          | TC     | GONXTBNK   |                                            |
| NXTSUPR  | AD     | SUPRCON    | # SET BNK 30 + INCR SUPR BNK AND CANCEL    |
|          | ADS    | SKEEP4     | # ERC BIT OF THE 37 TO 40 ADVANCE.         |
| GONXTBNK | CCS    | SKEEP7     |                                            |
|          | TC     | COMMFY     |                                            |
|          | CA     | S+1        |                                            |
|          | TC     | FXFX       |                                            |
|          | CA     | SBIT7      | # HAS TO BE LARGER THAN NO OF FXSW BANKS.  |
|          | TC     | COMMFY     |                                            |
| SOPTION  | CA     | SKEEP4     |                                            |
|          | MASK   | HI5        | # = BANK BITS                              |
|          | TC     | LEFT5      |                                            |
|          | TS     | L          | # BANK NUMBER BEFORE SUPER BANK            |

142THE

1

# SUBROUTINE TO UPDATE THE PROGRAM NUMBER DISPLAY ON THE DSKY.

COUNT\* \$\$/PHASE  
BLOCK 02  
SETLOC FFTAG1

BANK

NEWMODEX INDEX Q # UPDATE MODREG. ENTRY FOR MODE IN FIXED.

CAF 0  
INCR QNEWMODEA TS MODREG # ENTRY FOR MODE IN A.  
MMDSPY CAF +3 # DISPLAY MAJOR MODE.  
PREBJUMP LXCH BBANK # PUTS BBANK IN LTCF BANKJUMP # PUTS Q INTO A  
CADR SETUPDSP

# RETURN TO CALLER +3 IF MODE = THAT AT CALLER +1. OTHERWISE RETURN TO CALLER +2.

CHECKMM INDEX Q

CS 0  
AD MODREG  
EXTENDBZF Q+2  
TCF Q+1 # NO MATCH

TCQ = Q+2 +1

BANK 14  
SETLOC PHASETAB  
BANKSETUPDSP COUNT\* \$\$/PHASE  
INHINT  
DXCH RUPTREG1 # SAVE CALLER'S RETURN 2CADR  
CAF PRIO30 # EITHER A TASK OR JOB CAN COME TO  
TC NOVAC # NEWMODE X  
EBANK= MODREG  
2CADR DSPMMJOBDXCH RUPTREG1  
RELINT  
DXCH Z # RETURN

DSPMMJOB EQUALS DSPMMJB

BLOCK 02  
SETLOC FFTAG1  
BANK

# PHASCHNG IS THE MAIN WAY OF MAKING PHASE CHANGES FOR RESTARTS. THERE ARE THREE FORMS OF PHASCHNG, KNOWN AS TYPE  
# A, TYPE B, AND TYPE C. THEY ARE ALL CALLED AS FOLLOWS, WHERE OCT XXXXX CONTAINS THE PHASE INFORMATION,

# TC PHASCHNG  
# OCT XXXXX

# TYPE A IS CONCERNED WITH FIXED PHASE CHANGES, THAT IS, PHASE INFORMATION THAT IS STORED PERMANENTLY. THESE  
# OPTIONS ARE, WHERE G STANDS FOR A GROUP AND .X FOR THE PHASE,

# G.0 INACTIVE, WILL NOT PERMIT A GROUP G RESTART

# G.1 WILL CAUSE THE LAST DISPLAY TO BE REACTIVATED, USED MAINLY IN MANNED FLIGHTS

# G.EVEN A DOUBLE TABLE RESTART, CAN CAUSE ANY COMBINATION OF TWO JOBS, TASKS, AND/OR  
# LONGCALL TO BE RESTARTED.

# G.ODD NOT .1 A SINGLE TABLE RESTART, CAN CAUSE EITHER A JOB, TASK, OR LONGCALL RESTART.

# THIS INFORMATION IS PUT INTO THE OCTAL WORD AFTER TC PHASCHNG AS FOLLOWS

# TLO OOP PPP PPP GGG

# WHERE EACH LETTER OR NUMBER STANDS FOR A BIT. THE G'S STAND FOR THE GROUP, OCTAL 1-7, THE P'S FOR THE PHASE,

# OCTAL 0 - 127. O'S MUST BE 0. IF ONE WISHES TO HAVE THE TBASE OF GROUP G TO BE SET AT THIS TIME,

# T IS SET TO 1, OTHERWISE IT IS SET TO 0. SIMILARLY IF ONE WISHES TO SET LONGBASE, THEN L IS SET TO 1, OTHERWISE

# IT IS SET TO 0. SOME EXAMPLES,

# TC PHASCHNG # THIS WILL CAUSE GROUP 3 TO BE SET TO 0,  
# OCT 00003 # MAKING GROUP 3 INACTIVE

# TC PHASCHNG # IF A RESTART OCCURS THIS WOULD CAUSE  
# OCT 00012 # GROUP 2 TO RESTART THE LAST DISPLAY

# TC PHASCHNG # THIS SETS THE TBASE OF GROUP 4 AND IN  
# OCT 40064 # CASE OF A RESTART WOULD START UP THE TWO  
# THINGS LOCATED IN THE DOUBLE 4.6 RESTART  
# LOCATION.

# TC PHASCHNG # THIS SETS LONGBASE AND UPON A RESTART  
# OCT 20135 # CAUSES 5.13 TO BE RESTARTED (SINCE  
# LONGBASE WAS SET THIS SINGLE ENTRY  
# SHOULD BE A LONGCALL)

# TC PHASCHNG # SINCE BOTH TBASE4 AND LONGBASE ARE SET,  
# OCT 60124 # 4.12 SHOULD CONTAIN BOTH A TASK AND A  
# LONGCALL TO BE RESTARTED

# TYPE C PHASCHNG CONTAINS THE VARIABLE TYPE OF PHASCHNG INFORMATION. INSTEAD OF THE INFORMATION BEING IN A  
# PERMANENT FORM, ONE STORES THE DESIRED RESTART INFORMATION IN A VARIABLE LOCATION. THE BITS ARE AS FOLLOWS,  
# TLO 1AD XXX CJW GGG

# WHERE EACH LETTER OR NUMBER STANDS FOR A BIT. THE G'S STAND FOR THE GROUP, OCTAL 1 - 7. IF THE RESTART IS TO  
# BE BY WAITLIST, W IS SET TO 1, IF IT IS A JOB, J IS SET TO 1, IF IT IS A LONGCALL, C IS SET TO 1. ONLY ONE OF  
# THESE THREE BITS MAY BE SET. X'S ARE IGNORED, 1 MUST BE 1, AND 0 MUST BE 0. AGAIN T STANDS FOR THE TBASE,



```
1 # AND L FOR LONGBASE. THE BITS A AND D ARE CONCERNED WITH THE VARIABLE INFORMATION. IF D IS SET TO 1, A PRIORITY
2 # OR DELTA TIME WILL BE READ FROM THE NEXT LOCATION AFTER THE OCTAL INFORMATION., IF THIS IS TO BE INDIRECT, THAT
3 # IS, THE NAME OF A LOCATION CONTAINING THE INFORMATION (DELTA TIME ONLY), THEN THIS IS GIVEN AS THE -GENADR OF
4 # THAT LOCATION WHICH CONTAINS THE DELTA TIME. IF THE OLD PRIORITY OR DELTA TIME IS TO BE USED, THAT WHICH IS
5 # ALREADY IN THE VARIABLE STORAGE, THEN D IS SET TO 0. NEXT THE A BIT IS USED. IF IT IS SET TO 0, THE ADDRESS
6 # THAT WOULD BE RESTARTED DURING A RESTART IS THE NEXT LOCATION AFTER THE PHASE INFORMATION, THAT IS, EITHER
7 # (TC PHASCHNG) +2 OR +3, DEPENDING ON WHETHER D HAD BEEN SET OR NOT. IF A IS SET TO 1, THEN THE ADDRESS THAT
8 # WOULD BE RESTARTED IS THE 2CADR THAT IS READ FROM THE NEXT TWO LOCATION. EXAMPLES,
9
10 # AD TC PHASCHNG # THIS WOULD CAUSE LOCATION AD +3 TO BE
11 # AD+1 OCT 05023 # RESTARTED BY GROUP THREE WITH A PRIORITY
12 # AD+2 OCT 23000 # OF 23. NOTE UPON RETURNING IT WOULD
13 # AD+3 # ALSO GO TO AD+3
14
15 # AD TC PHASCHNG # GROUP 1 WOULD CAUSE CALLCALL TO BE
16 # AD+1 OCT 27441 # BE STARTED AS A LONGCALL FROM THE TIME
17 # AD+2 -GENADR DELTIME # STORED IN LONGBASE (LONGBASE WAS SET) BY
18 # AD+3 2CADR CALLCALL # A DELTA TIME STORED IN DELTIME. THE
19 # AD+4 # BBCON OF THE 2CADR SHOULD CONTAIN THE E
20 # AD+5 # BANK OF DELTIME. PHASCHNG RETURNS TO
21 # # LOCATION AD+5
22
23 # NOTE THAT IF A VARIABLE PRIORITY IS GIVEN FOR A JOB, THE JOB WILL BE RESTARTED AS A NOVAC IF THE PRIORITY IS
24 # NEGATIVE, AS A FINDVAC IF THE PRIORITY IS POSITIVE.
25
26 # TYPE B PHASCHNG IS A COMBINATION OF VARIABLE AND FIXED PHASE CHANGES. IT WILL START UP A JOB AS INDICATED
27 # BELOW AND ALSO START UP ONE FIXED RESTART, THAT IS EITHER AN G.1 OR A G.ODD OR THE FIRST ENTRY OF G.EVEN
28 # DOUBLE ENTRY. THE BIT INFORMATION IS AS FOLLOW,
29 # TL1 DAP PPP PPP GGG
30 # WHERE EACH LETTER OR NUMBER STANDS FOR A BIT. THE G'S STAND FOR THE GROUP, OCTAL 1 - 7, THE P'S FOR THE FIXED
31 # PHASE INFORMATION, OCTAL 0 - 127. 1 MUST BE 1. AND AGAIN T STANDS FOR THE TBASE AND L FOR LONGBASE. D THIS
32 # TIME STANDS ONLY FOR PRIORITY SINCE THIS WILL BE CONSIDERED A JOB, AND IT MUST BE GIVEN DIRECTLY IF GIVEN.
33 # AGAIN A STANDS FOR THE ADDRESS OF THE LOCATION TO BE RESTARTED, 1 IF THE 2CADR IS GIVEN, OR 0 IF IT IS TO BE
34 # THE NEXT LOCATION. (THE RETURN LOCATION OF PHASCHNG) EXAMPLES,
35 # AD TC PHASCHNG # TBASE IS SET AND A RESTART CAUSE GROUP 3
36 # AD+1 OCT 56043 # TO START THE JOB AJOBAJOB WITH PRIORITY
37 # AD+2 OCT 31000 # 31 AND THE FIRST ENTRY OF 3.4SPOT (WE CAN
38 # AD+3 2CADR AJOBAJOB # ASSUME IT IS A TASK SINCE WE SET TBASE3)
39 # AD+4 # UPON RETURN FROM PHASCHNG CONTROL WOULD
40 # AD+5 # GO TO AD+5
41
42 # AD TC PHASCHNG # UPON A RESTART THE LAST DISPLAY WOULD BE
43 # AD+1 OCT 10015 # RESTARTED AND A JOB WITH THE PREVIOUSLY
44 # AD+2 # STORED PRIORITY WOULD BE BEGUN AT AD+2
45 # # BY MEANS OF GROUP 5
```



|    |                                                                                                                   |        |     |     |     |     |     |    |
|----|-------------------------------------------------------------------------------------------------------------------|--------|-----|-----|-----|-----|-----|----|
| 1  | # THE NOVAC-FINDVAC CHOICE FOR JOBS HOLDS HERE ALSO -- NEGATIVE PRIORITY CAUSES A NOVAC CALL, POSITIVE A FINDVAC. |        |     |     |     |     |     | 1  |
| 2  |                                                                                                                   |        |     |     |     |     |     | 2  |
| 3  |                                                                                                                   |        |     |     |     |     |     | 3  |
| 4  | # SUMMARY OF BITS:                                                                                                |        |     |     |     |     |     | 4  |
| 5  | #                                                                                                                 | TYPE A | TL0 | OOP | PPP | PPP | GGG | 5  |
| 6  | #                                                                                                                 | TYPE B | TL1 | DAP | PPP | PPP | GGG | 6  |
| 7  | #                                                                                                                 | TYPE C | TL0 | 1AD | XXX | CJW | GGG | 7  |
| 8  |                                                                                                                   |        |     |     |     |     |     | 8  |
| 9  |                                                                                                                   |        |     |     |     |     |     | 9  |
| 10 |                                                                                                                   |        |     |     |     |     |     | 10 |
| 11 |                                                                                                                   |        |     |     |     |     |     | 11 |
| 12 |                                                                                                                   |        |     |     |     |     |     | 12 |
| 13 |                                                                                                                   |        |     |     |     |     |     | 13 |
| 14 |                                                                                                                   |        |     |     |     |     |     | 14 |
| 15 |                                                                                                                   |        |     |     |     |     |     | 15 |
| 16 |                                                                                                                   |        |     |     |     |     |     | 16 |
| 17 |                                                                                                                   |        |     |     |     |     |     | 17 |
| 18 |                                                                                                                   |        |     |     |     |     |     | 18 |
| 19 |                                                                                                                   |        |     |     |     |     |     | 19 |
| 20 |                                                                                                                   |        |     |     |     |     |     | 20 |
| 21 |                                                                                                                   |        |     |     |     |     |     | 21 |
| 22 |                                                                                                                   |        |     |     |     |     |     | 22 |
| 23 |                                                                                                                   |        |     |     |     |     |     | 23 |
| 24 |                                                                                                                   |        |     |     |     |     |     | 24 |
| 25 |                                                                                                                   |        |     |     |     |     |     | 25 |
| 26 |                                                                                                                   |        |     |     |     |     |     | 26 |
| 27 |                                                                                                                   |        |     |     |     |     |     | 27 |
| 28 |                                                                                                                   |        |     |     |     |     |     | 28 |
| 29 |                                                                                                                   |        |     |     |     |     |     | 29 |
| 30 |                                                                                                                   |        |     |     |     |     |     | 30 |
| 31 |                                                                                                                   |        |     |     |     |     |     | 31 |
| 32 |                                                                                                                   |        |     |     |     |     |     | 32 |
| 33 |                                                                                                                   |        |     |     |     |     |     | 33 |
| 34 |                                                                                                                   |        |     |     |     |     |     | 34 |
| 35 |                                                                                                                   |        |     |     |     |     |     | 35 |
| 36 |                                                                                                                   |        |     |     |     |     |     | 36 |
| 37 |                                                                                                                   |        |     |     |     |     |     | 37 |
| 38 |                                                                                                                   |        |     |     |     |     |     | 38 |
| 39 |                                                                                                                   |        |     |     |     |     |     | 39 |
| 40 |                                                                                                                   |        |     |     |     |     |     | 40 |
| 41 |                                                                                                                   |        |     |     |     |     |     | 41 |
| 42 |                                                                                                                   |        |     |     |     |     |     | 42 |
| 43 |                                                                                                                   |        |     |     |     |     |     | 43 |
| 44 |                                                                                                                   |        |     |     |     |     |     | 44 |
| 45 |                                                                                                                   |        |     |     |     |     |     | 45 |
| 46 |                                                                                                                   |        |     |     |     |     |     | 46 |
| 47 |                                                                                                                   |        |     |     |     |     |     | 47 |
| 48 |                                                                                                                   |        |     |     |     |     |     | 48 |
| 49 |                                                                                                                   |        |     |     |     |     |     | 49 |
| 50 |                                                                                                                   |        |     |     |     |     |     | 50 |
| 51 |                                                                                                                   |        |     |     |     |     |     | 51 |
| 52 |                                                                                                                   |        |     |     |     |     |     | 52 |
| 53 |                                                                                                                   |        |     |     |     |     |     | 53 |
| 54 |                                                                                                                   |        |     |     |     |     |     | 54 |
| 55 |                                                                                                                   |        |     |     |     |     |     | 55 |
| 56 |                                                                                                                   |        |     |     |     |     |     | 56 |
| 57 |                                                                                                                   |        |     |     |     |     |     | 57 |
| 58 |                                                                                                                   |        |     |     |     |     |     | 58 |
| 59 |                                                                                                                   |        |     |     |     |     |     | 59 |
| 60 |                                                                                                                   |        |     |     |     |     |     | 60 |

```
2PHSCHNG IS USED WHEN ONE WISHES TO START UP A GROUP OR CHANGE A GROUP WHILE UNDER THE CONTROL OF A DIFFERENT
GROUP. FOR EXAMPLE, CHANGE THE PHASE OF GROUP 3 WHILE THE PORTION OF THE PROGRAM IS UNDER GROUP 5. ALL 2PHSCHNG
CALLS ARE MADE IN THE FOLLOWING MANNER,
TC 2PHSCHNG
OCT XXXXX
OCT YYYYY
WHERE OCT XXXXX MUST BE OF TYPE A AND OCT YYYYY MAY BE OF EITHER TYPE A OR TYPE B OR TYPE C. THERE IS ONE
DIFFERENCE --- NOTE: IF LONGBASE IS TO BE SET THIS INFORMATION IS GIVEN IN THE OCT YYYYY INFORMATION, IT WILL
BE DISREGARDED IF GIVEN WITH THE OCT XXXXX INFORMATION. A COUPLE OF EXAMPLES MAY HELP,
AD TC 2PHACHNG # SET TBASE3 AND IF A RESTART OCCURS START
AD+1 OCT 40083 # THE TWO ENTRIES IN 3.8 TABLE LOCATION
AD+2 OCT 05025 # THIS IS OF TYPE C, SET THE JOB TO BE
AD+3 OCT 18000 # TO BE LOCATION AD+4, WITH A PRIORITY 18,
AD+4 # FOR GROUP 5 PHASE INFORMATION.
```

```
SBANK= PINSUPER
```

```
2PHSCHNG COUNT* $$/PHASE
 INHINT
 NDX Q
 CA 0
 INCR Q
 TS TEMPP2

 MASK OCT7
 DOUBLE
 TS TEMPG2

 CA TEMPP2
 MASK OCT17770 # NEED ONLY 1770, BUT WHY GET A NEW CONST.
 EXTEND
 MP BIT12
 XCH TEMPP2

 MASK BIT15
 TS TEMPSW2 # INDICATES WHETHER TO SET TBASE OR NOT

 INDEX Q
 CA 0
 INCR Q
 TS TEMPSW

 TCF PHASJUMP

PHASCHNG INHINT
 INDEX Q
 CA 0
 INCR Q
PHSCHNGA INHINT
 INHINT
```

```
FIRST OCTAL PARAMETER IN A.
```

|          |        |             |                                       |
|----------|--------|-------------|---------------------------------------|
|          | TS     | TEMPSW      |                                       |
|          | CA     | ONE         |                                       |
| PHASJUMP | TS     | TEMPSW2     |                                       |
|          | EXTEND |             |                                       |
|          | DCA    | ADRPCHN2    | # OFF TO SWITCHED BANK                |
|          | DTCB   |             |                                       |
|          | EBANK= | LST1        |                                       |
| ADRPCHN2 | 2CADR  | PHSCHNG2    |                                       |
| ONEORTWO | LXCH   | TEMPBBCN    |                                       |
|          | LXCH   | BBANK       |                                       |
|          | LXCH   | TEMPBBCN    |                                       |
|          | MASK   | OCT14000    | # SEE WHAT KIND OF PHASE CHANGE IT IS |
|          | CCS    | A           |                                       |
|          | TCF    | CHECKB      | # IT IS OF TYPE `B'.                  |
|          | CA     | TEMPP       |                                       |
|          | MASK   | BIT7        |                                       |
|          | CCS    | A           | # SHALL WE USE THE OLD PRIORITY       |
|          | TCF    | GETPRIO     | # NO GET A NEW PRIORITY (OR DELTA T)  |
| OLDPRIO  | NDX    | TEMPPG      | # USE THE OLD PRIORITY (OR DELTA T)   |
|          | CA     | PHSPRDT1 -2 |                                       |
|          | TS     | TEMPPR      |                                       |
| CON1     | CA     | TEMPP       | # SEE IF A 2CADR IS GIVEN             |
|          | MASK   | BIT8        |                                       |
|          | CCS    | A           |                                       |
|          | TCF    | GETNEWNM    |                                       |
|          | CA     | Q           |                                       |
|          | TS     | TEMPNM      |                                       |
|          | CA     | BB          |                                       |
|          | EXTEND |             | # PICK UP USER'S SUPERBANK            |
|          | ROR    | SUPERBNK    |                                       |
|          | TS     | TEMPBB      |                                       |
| TOCON2   | CA     | CON2ADR     | # BACK TO SWITCHED BANK               |
|          | LXCH   | TEMPBBCN    |                                       |
|          | DTCB   |             |                                       |
| CON2ADR  | GENADR | CON2        |                                       |
| GETPRIO  | NDX    | Q           | # DON'T CARE IF DIRECT OR INDIRECT    |
|          | CA     | 0           | # LEAVE THAT DECISION TO RESTARTS     |
|          | INCR   | Q           | # OBTAIN RETURN ADDRESS               |

|    |                         |        |            |    |
|----|-------------------------|--------|------------|----|
| 1  |                         |        |            | 1  |
| 2  |                         | TCF    | CON1 -1    | 2  |
| 3  |                         |        |            | 3  |
| 4  | GETNEWNM                | EXTEND |            | 4  |
| 5  |                         | INDEX  | Q          | 5  |
| 6  |                         | DCA    | 0          | 6  |
| 7  |                         | DXCH   | TEMPNM     | 7  |
| 8  |                         | CA     | TWO        | 8  |
| 9  |                         | ADS    | Q          | 9  |
| 10 | # OBTAIN RETURN ADDRESS |        |            | 10 |
| 11 |                         | TCF    | TOCON2     | 11 |
| 12 |                         |        |            | 12 |
| 13 | OCT14000                | EQUALS | PRI014     | 13 |
| 14 | TEMPG                   | EQUALS | ITEMP1     | 14 |
| 15 | TEMPP                   | EQUALS | ITEMP2     | 15 |
| 16 | TEMPNM                  | EQUALS | ITEMP3     | 16 |
| 17 | TEMPBB                  | EQUALS | ITEMP4     | 17 |
| 18 | TEMPSW                  | EQUALS | ITEMP5     | 18 |
| 19 | TEMPSW2                 | EQUALS | ITEMP6     | 19 |
| 20 | TEMPPR                  | EQUALS | RUPTREG1   | 20 |
| 21 | TEMPG2                  | EQUALS | RUPTREG2   | 21 |
| 22 | TEMPP2                  | EQUALS | RUPTREG3   | 22 |
| 23 |                         |        |            | 23 |
| 24 | TEMPBBCN                | EQUALS | RUPTREG4   | 24 |
| 25 | BB                      | EQUALS | BBANK      | 25 |
| 26 |                         |        |            | 26 |
| 27 |                         | BANK   | 14         | 27 |
| 28 |                         | SETLOC | PHASETAB   | 28 |
| 29 |                         | BANK   |            | 29 |
| 30 |                         |        |            | 30 |
| 31 |                         | EBANK= | PHSNAME1   | 31 |
| 32 |                         | COUNT* | \$\$/PHASE | 32 |
| 33 | PHSCHNG2                | LXCH   | TEMPBBCN   | 33 |
| 34 |                         | CA     | TEMPSW     | 34 |
| 35 |                         | MASK   | OCT7       | 35 |
| 36 |                         | DOUBLE |            | 36 |
| 37 |                         | TS     | TEMPG      | 37 |
| 38 |                         |        |            | 38 |
| 39 |                         | CA     | TEMPSW     | 39 |
| 40 |                         | MASK   | OCT17770   | 40 |
| 41 |                         | EXTEND |            | 41 |
| 42 |                         | MP     | BIT12      | 42 |
| 43 |                         | TS     | TEMPP      | 43 |
| 44 |                         |        |            | 44 |
| 45 |                         | CA     | TEMPSW     | 45 |
| 46 |                         | MASK   | OCT60000   | 46 |
| 47 |                         | XCH    | TEMPSW     | 47 |
| 48 |                         | MASK   | OCT14000   | 48 |
| 49 |                         | CCS    | A          | 49 |
| 50 |                         |        |            | 50 |
| 51 |                         |        |            | 51 |
| 52 |                         |        |            | 52 |
| 53 |                         |        |            | 53 |
| 54 |                         |        |            | 54 |
| 55 |                         |        |            | 55 |
| 56 |                         |        |            | 56 |
| 57 |                         |        |            | 57 |
| 58 |                         |        |            | 58 |
| 59 |                         |        |            | 59 |
| 60 |                         |        |            | 60 |

|          |        |            |                                          |
|----------|--------|------------|------------------------------------------|
|          | TCF    | ONEORTWO   |                                          |
|          | CA     | TEMPP      | # START STORING THE PHASE INFORMATION    |
|          | NDX    | TEMPG      |                                          |
|          | TS     | PHASE1 -2  |                                          |
| BELOW1   | CCS    | TEMPSW2    | # IS IT A PHASCHNG OR A 2PHSCHNG         |
|          | TCF    | BELOW2     | # IT'S A PHASCHNG                        |
|          | TCF    | +1         | # IT'S A 2PHSCHNG                        |
|          | CS     | TEMPP2     |                                          |
|          | LXCH   | TEMPP2     |                                          |
|          | NDX    | TEMPG2     |                                          |
|          | DXCH   | -PHASE1 -2 |                                          |
|          | CCS    | TEMPSW2    |                                          |
|          | NOOP   |            | # CAN'T GET HERE                         |
|          | TCF    | BELOW2     |                                          |
|          | CS     | TIME1      |                                          |
|          | NDX    | TEMPG2     |                                          |
|          | TS     | TBASE1 -2  |                                          |
| BELOW2   | CCS    | TEMPSW     | # SEE IF WE SHOULD SET TBASE OR LONGBASE |
|          | TCF    | BELOW3     | # SET LONGBASE ONLY                      |
|          | TCF    | BELOW4     | # SET NEITHER                            |
|          | CS     | TIME1      | # SET TBASE TO BEGIN WITH                |
|          | NDX    | TEMPG      |                                          |
|          | TS     | TBASE1 -2  |                                          |
|          | CA     | TEMPSW     | # SHALL WE NOW SET LONGBASE              |
|          | AD     | BIT14COM   |                                          |
|          | CCS    | A          |                                          |
|          | NOOP   |            | # ***** CAN'T GET HERE *****             |
| BIT14COM | OCT    | 17777      | # ***** CAN'T GET HERE *****             |
|          | TCF    | BELOW4     | # NO WE NEED ONLY SET TBASE              |
| BELOW3   | EXTEND |            | # SET LONGBASE                           |
|          | DCA    | TIME2      |                                          |
|          | DXCH   | LONGBASE   |                                          |
| BELOW4   | CS     | TEMPP      | # AND STORE THE FINAL PART OF THE PHASE  |
|          | NDX    | TEMPG      |                                          |
|          | TS     | -PHASE1 -2 |                                          |
|          | CA     | Q          |                                          |
|          | LXCH   | TEMPBBCN   |                                          |
|          | RELINT |            |                                          |
|          | DTCB   |            |                                          |



|    |        |        |             |                                           |    |
|----|--------|--------|-------------|-------------------------------------------|----|
| 1  |        |        |             |                                           | 1  |
| 2  | CON2   | LXCH   | TEMPBBCN    |                                           | 2  |
| 3  |        |        |             |                                           | 3  |
| 4  |        | CA     | TEMPP       |                                           | 4  |
| 5  |        | NDX    | TEMPG       |                                           | 5  |
| 6  |        | TS     | PHASE1 -2   |                                           | 6  |
| 7  |        |        |             |                                           | 7  |
| 8  |        | CA     | TEMPPR      |                                           | 8  |
| 9  |        | NDX    | TEMPG       |                                           | 9  |
| 10 |        | TS     | PHSPRDT1 -2 |                                           | 10 |
| 11 |        |        |             |                                           | 11 |
| 12 |        | EXTEND |             |                                           | 12 |
| 13 |        | DCA    | TEMPNM      |                                           | 13 |
| 14 |        | NDX    | TEMPG       |                                           | 14 |
| 15 |        | DXCH   | PHSNAME1 -2 |                                           | 15 |
| 16 |        |        |             |                                           | 16 |
| 17 |        | TCF    | BELOW1      |                                           | 17 |
| 18 |        |        |             |                                           | 18 |
| 19 |        | BLOCK  | 03          |                                           | 19 |
| 20 |        | SETLOC | FFTAG6      |                                           | 20 |
| 21 |        | BANK   |             |                                           | 21 |
| 22 |        |        |             |                                           | 22 |
| 23 |        | COUNT* | \$\$/PHASE  |                                           | 23 |
| 24 | CHECKB | MASK   | BIT12       | # SINCE THIS IS OF TYPE B, THIS BIT WOULD | 24 |
| 25 |        | CCS    | A           | # BE HERE IF WE ARE TO GET A NEW PRIORITY | 25 |
| 26 |        | TCF    | GETPRIO     | # IT IS, SO GET NEW PRIORITY              | 26 |
| 27 |        |        |             |                                           | 27 |
| 28 |        | TCF    | OLDPRIO     | # IT ISN'T, USE THE OLD PRIORITY.         | 28 |
| 29 |        |        |             |                                           | 29 |
| 30 |        |        |             |                                           | 30 |
| 31 |        |        |             |                                           | 31 |
| 32 |        |        |             |                                           | 32 |
| 33 |        |        |             |                                           | 33 |
| 34 |        |        |             |                                           | 34 |
| 35 |        |        |             |                                           | 35 |
| 36 |        |        |             |                                           | 36 |
| 37 |        |        |             |                                           | 37 |
| 38 |        |        |             |                                           | 38 |
| 39 |        |        |             |                                           | 39 |
| 40 |        |        |             |                                           | 40 |
| 41 |        |        |             |                                           | 41 |
| 42 |        |        |             |                                           | 42 |
| 43 |        |        |             |                                           | 43 |
| 44 |        |        |             |                                           | 44 |
| 45 |        |        |             |                                           | 45 |
| 46 |        |        |             |                                           | 46 |
| 47 |        |        |             |                                           | 47 |
| 48 |        |        |             |                                           | 48 |
| 49 |        |        |             |                                           | 49 |
| 50 |        |        |             |                                           | 50 |
| 51 |        |        |             |                                           | 51 |
| 52 |        |        |             |                                           | 52 |
| 53 |        |        |             |                                           | 53 |
| 54 |        |        |             |                                           | 54 |
| 55 |        |        |             |                                           | 55 |
| 56 |        |        |             |                                           | 56 |
| 57 |        |        |             |                                           | 57 |
| 58 |        |        |             |                                           | 58 |
| 59 |        |        |             |                                           | 59 |
| 60 |        |        |             |                                           | 60 |

```
1
2 BANK 01
3 SETLOC RESTART
4 BANK
5
6 EBANK= PHSNAME1 # GOPROG MUST SWITCH TO THIS EBANK
7
8 COUNT* $$/RSROU
9 RESTARTS CA MPAC +5 # GET GROUP NUMBER -1
10 DOUBLE
11 TS TEMP2G # SAVE FOR INDEXING
12
13 CA PHS2CADR # SET UP EXIT IN CASE IT IS AN EVEN
14 TS TEMPSWCH # TABLE PHASE
15
16 CA RTRNCADR # TO SAVE TIME ASSUME IT WILL GET NEXT
17 TS GOLOC +2 # GROUP AFTER THIS
18
19 CA TEMPPHS
20 MASK OCT1400
21 CCS A # IS IT A VARIABLE OR TABLE RESTART
22 TCF ITSAVAR # IT:S A VARIABLE RESTART
23
24 GETPART2 CCS TEMPPHS # IS IT AN X.1 RESTART
25 CCS A
26 TCF ITSATBL # NO, ITS A TABLE RESTART
27
28 CA PRI014 # IT IS AN X.1 RESTART, THEREFORE START
29 TC FINDVAC # THE DISPLAY RESTART JOB
30 EBANK= LST1
31 2CADR INITDSP
32
33 TC RTRNCADR # FINISHED WITH THIS GROUP, GET NEXT ONE
34
35 ITSAVAR MASK OCT1400 # IS IT TYPE B ?
36 CCS A
37 TCF ITSLIKEB # YES,IT IS TYPE B
38
39 EXTEND # STORE THE JOB (OR TASK) 2CADR FOR EXIT
40 NDX TEMP2G
41 DCA PHSNAME1
42 DXCH GOLOC
43
44 CA TEMPPHS # SEE IF THIS IS A JOB, TASK, OR A LONGCAL
45 MASK OCT7
46 AD MINUS2
47 CCS A
48 TCF ITSLNGCL # ITS A LONGCALL
49
50 RTRNCADR TC SWRETURN # CANT GET HERE
51
52
53
54
55
56
57
58
59
60
```



```
1
2 TCF ITS await
3
4 TCF ITS a JOB # ITS A JOB
5
6 ITS await CA WTLTCADR # SET UP WAITLIST CALL
7 TS GOLOC -1
8
9 NDX TEMP2G # DIRECTLY STORED
10
11 TIMETEST CA PHSPRDT1
12 CCS A # IS IT AN IMMEDIATE RESTART
13 INCR A # NO.
14
15 TCF FINDTIME # FIND OUT WHEN IT SHOULD BEGIN
16
17 TCF ITSINDIR # STORED INDIRECTLY
18
19 TCF IMEDIATE # IT WANTS AN IMMEDIATE RESTART
20
21 # ***** THIS MUST BE IN FIXED FIXED *****
22
23 BLOCK 02
24 SETLOC FFTAG2
25 BANK
26
27 ITSINDIR COUNT* $$/RSROU
28 LXCH GOLOC +1 # GET THE CORRECT E BANK IN CASE THIS IS
29 LXCH BB # SWITCHED ERRASIBLE
30
31 NDX A # GET THE TIME INDIRECTLY
32 CA 1
33
34 LXCH BB # RESTORE THE BB AND GOLOC
35 LXCH GOLOC +1
36
37 TCF FINDTIME # FIND OUT WHEN IT SHOULD BEGIN
38
39 # ***** YOU MAY RETURN TO SWITCHED FIXED *****
40
41 BANK 01
42 SETLOC RESTART
43 BANK
44
45 FINDTIME COUNT* $$/RSROU
46 COM L # MAKE NEGATIVE SINCE IT WILL BE SUBTRACTD
47 TS L # AND SAVE
48
49 NDX TEMP2G
50 CS TBASE1
51 EXTEND
52
53 SU TIME1
54 CCS A
55 COM
```

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
|          | AD     | OCT37776 |                                           |
|          | AD     | ONE      |                                           |
|          | AD     | L        |                                           |
|          | CCS    | A        |                                           |
|          | CA     | ZERO     |                                           |
|          | TCF    | +2       |                                           |
|          | TCF    | +1       |                                           |
| IMEDIATE | AD     | ONE      |                                           |
| ITSLIKEB | TC     | GOLOC -1 |                                           |
|          | CA     | RTRNCADR | # TYPE B, SO STORE RETURN IN              |
|          | TS     | TEMPSWCH | # TEMPSWCH IN CASE OF AN EVEN PHASE       |
|          | CA     | PRT2CADR | # SET UP EXIT TO GET TABLE PART OF THIS   |
|          | TS     | GOLOC +2 | # VARIABLE TYPE OF PHASE                  |
|          | CA     | TEMPPHS  | # MAKE THE PHASE LOOK RIGHT FOR THE TABLE |
|          | MASK   | OCT177   | # PART OF THIS VARIABLE PHASE             |
|          | TS     | TEMPPHS  |                                           |
|          | EXTEND |          |                                           |
|          | NDX    | TEMP2G   | # OBTAIN THE JOB:S 2CADR                  |
|          | DCA    | PHSNAME1 |                                           |
|          | DXCH   | GOLOC    |                                           |
| ITSAJOB  | NDX    | TEMP2G   | # NOW ADD THE PRIORITY AND LET:S GO       |
|          | CA     | PHSPRDT1 |                                           |
| CHKNOVAC | TS     | GOLOC -1 | # SAVE PRIO UNTIL WE SEE IF ITS           |
|          | EXTEND |          | # A FINDVAC OR A NOVAC                    |
|          | BZMF   | ITSNOVAC |                                           |
|          | CAF    | FVACCADR | # POSITIVE, SET UP FINDVAC CALL.          |
|          | XCH    | GOLOC -1 | # PICK UP PRIO,                           |
|          | TC     | GOLOC -1 | # AND GO                                  |
| ITSNOVAC | CAF    | NOVACADR | # NEGATIVE,                               |
|          | XCH    | GOLOC -1 | # SET UP NOVAC CALL,                      |
|          | COM    |          | # CORRECT PRIO,                           |
|          | TC     | GOLOC -1 | # AND GO                                  |
| ITSATBL  | TS     | CYR      | # FIND OUT IF THE PHASE IS ODD OR EVEN    |
|          | CCS    | CYR      |                                           |
|          | TCF    | +1       | # IT:S EVEN                               |
|          | TCF    | ITSEVEN  |                                           |
|          | CA     | RTRNCADR | # IN CASE THIS IS THE SECOND PART OF A    |
|          | TS     | GOLOC +2 | # TYPE B RESTART, WE NEED PROPER EXIT     |
|          | CA     | TEMPPHS  | # SET UP POINTER FOR FINDING OUR PLACE IN |
|          | TS     | SR       | # THE RESTART TABLES                      |
|          | AD     | SR       |                                           |

```
1
2 NDX TEMP2G
3 AD SIZETAB +1
4 TS POINTER
5
6 CONTBL2 EXTEND # FIND OUT WHAT:S IN THE TABLE
7 NDX POINTER
8 DCA CADRTAB # GET THE 2CADR
9
10 LXCH GOLOC +1 # STORE THE BB INFORMATION
11
12 CCS A # IS IT A JOB OR IS IT TIMED
13 INCR A # POSITIVE. MUST BE A JOB
14 TCF ITSAJOB2
15
16 INCR A # MUST BE EITHER A WAITLIST OR LONGCALL
17 TS GOLOC # LET-S STORE THE CORRECT CADR
18
19 CA WTLTCADR # SET UP OUR EXIT TO WAITLIST
20 TS GOLOC -1
21
22 CA GOLOC +1 # NOW FIND OUT IF IT IS A WAITLIST CALL
23 MASK BIT10 # THIS SHOULD BE ONE IF WE HAVE -BB
24 CCS A # FOR THAT MATTER SO SHOULD BE BITS 9,8,7,
25 # 6,5, AND LAST BUT NOT LEAST (PERHAPS NOT
26 # IN IMPORTANCE ANYWAY. BIT 4
27 TCF ITSWTLST # IT IS A WAITLIST CALL
28
29 NDX POINTER # OBTAIN THE ORIGINAL DELTA T
30 CA PRDTTAB # ADDRESS FOR THIS LONGCALL
31
32 TCF ITSLGCL1 # NOW GO GET THE DELTA TIME
33
34 # ***** THIS MUST BE IN FIXED FIXED *****
35
36 BLOCK 02
37 SETLOC FFTAG2
38 BANK
39
40 COUNT* $$/RSROU
41 ITSLGCL1 LXCH GOLOC +1 # OBTAIN THE CORRECT E BANK
42 LXCH BB
43 LXCH GOLOC +1 # AND PRESERVE OUR E AND F BANKS
44
45 EXTEND # GET THE DELTA TIME
46 NDX A
47 DCA 0
48
49 LXCH GOLOC +1 # RESTORE OUR E AND F BANK
50 LXCH BB # RESTORE THE TASKS E AND F BANKS
51 LXCH GOLOC +1 # AND PRESERVE OUR L
```

```
1 TCF ITSLGCL2 # NOT LET:S PROCESS THIS LONGCALL
2
3
4 # ***** YOU MAY RETURN TO SWITCHED FIXED *****
5
6 BANK 01
7 SETLOC RESTART
8 BANK
9
10 COUNT* $$/RSROU
11 ITSLGCL2 DXCH LONGTIME
12
13 EXTEND
14 DCS TIME2 # CALCULATE TIME LEFT
15 DAS LONGTIME
16
17 EXTEND
18 DCA LONGBASE
19 DAS LONGTIME
20
21 CCS LONGTIME # FIND OUT HOW THIS SHOULD BE RESTARTED
22 TCF LONGCLCL
23 TCF +2
24 TCF IMEDIATE -3
25 CCS LONGTIME +1
26 TCF LONGCLCL
27 NOOP
28 TCF IMEDIATE -3 # CAN:T GET HERE *****
29 TCF IMEDIATE
30
31 LONGCLCL CA LGCLCADR # WE WILL GO TO LONGCALL
32 TS GOLOC -1
33
34 EXTEND
35 DCA LONGTIME # PREPARE OUR ENTRY TO LONGCALL
36 TC GOLOC -1
37
38 ITSLNGCL CA WTLTCADR # ASSUME IT WILL GO TO WAITLIST
39 TS GOLOC -1
40
41 NDX TEMP2G
42 CS PHSPRDT1 # GET THE DELTA T ADDRESS
43
44 TCF ITSLGCL1 # NOW GET THE DELTA TIME
45
46 ITSWTLST CS GOLOC +1 # CORRECT THE BBCON INFORMATION
47 TS GOLOC +1
48
49 NDX POINTER # GET THE DT AND FIND OUT IF IT WAS STORED
50 CA PRDTTAB # DIRECTLY OR INDIRECTLY
51
52 TCF TIMETEST # FIND OUT HOW THE TIME IS STORED
53
54
55
56
57
58
59
60
```

|    |          |        |           |                                                       |    |
|----|----------|--------|-----------|-------------------------------------------------------|----|
| 1  |          |        |           |                                                       | 1  |
| 2  | ITSAJOB2 | XCH    | GOLOC     | # STORE THE CADR                                      | 2  |
| 3  |          |        |           |                                                       | 3  |
| 4  |          | NDX    | POINTER   | # ADD THE PRIORITY AND LET:S GO                       | 4  |
| 5  |          | CA     | PRDTTAB   |                                                       | 5  |
| 6  |          |        |           |                                                       | 6  |
| 7  |          | TCF    | CHKNOVAC  |                                                       | 7  |
| 8  |          |        |           |                                                       | 8  |
| 9  | ITSEVEN  | CA     | TEMPSWCH  | # SET UP FOR EITHER THE SECOND PART OF THE            | 9  |
| 10 |          | TS     | GOLOC +2  | # TABLE, OR A RETURN FOR THE NEXT GROUP               | 10 |
| 11 |          |        |           |                                                       | 11 |
| 12 |          | NDX    | TEMP2G    | # SET UP POINTER FOR OUR LOCATION WITHIN              | 12 |
| 13 |          | CA     | SIZETAB   | # THE TABLE                                           | 13 |
| 14 |          | AD     | TEMPPHS   | # THIS MAY LOOK BAD BUT LET:S SEE YOU DO              | 14 |
| 15 |          | AD     | TEMPPHS   | # BETTER IN TIME OR NUMBERR OF LOCATIONS              | 15 |
| 16 |          |        |           |                                                       | 16 |
| 17 |          | AD     | TEMPPHS   |                                                       | 17 |
| 18 |          | TS     | POINTER   |                                                       | 18 |
| 19 |          |        |           |                                                       | 19 |
| 20 |          | TCF    | CONTBL2   | # NOW PROCESS WHAT IS IN THE TABLE                    | 20 |
| 21 | PHSPART2 | CA     | THREE     | # SET THE POINTER FOR THE SECOND HALF OF              | 21 |
| 22 |          | ADS    | POINTER   | # THE TABLE                                           | 22 |
| 23 |          |        |           |                                                       | 23 |
| 24 |          | CA     | RTRNCADR  | # THIS WILL BE OUR LAST TIME THROUGH THE              | 24 |
| 25 |          | TS     | GOLOC +2  | # EVEN TABLE , SO AFTER IT GET THE NEXT               | 25 |
| 26 |          |        |           |                                                       | 26 |
| 27 |          | TCF    | CONTBL2   | # GROUP<br># SO LET:S GET THE SECOND ENTRY IN THE TBL | 27 |
| 28 |          |        |           |                                                       | 28 |
| 29 | TEMPPHS  | EQUALS | MPAC      |                                                       | 29 |
| 30 | TEMP2G   | EQUALS | MPAC +1   |                                                       | 30 |
| 31 | POINTER  | EQUALS | MPAC +2   |                                                       | 31 |
| 32 | TEMPSWCH | EQUALS | MPAC +3   |                                                       | 32 |
| 33 | GOLOC    | EQUALS | VAC5 +20D |                                                       | 33 |
| 34 | MINUS2   | EQUALS | NEG2      |                                                       | 34 |
| 35 | OCT177   | EQUALS | LOW7      |                                                       | 35 |
| 36 |          |        |           |                                                       | 36 |
| 37 | PHS2CADR | GENADR | PHSPART2  |                                                       | 37 |
| 38 | PRT2CADR | GENADR | GETPART2  |                                                       | 38 |
| 39 | LGCLCADR | GENADR | LONGCALL  |                                                       | 39 |
| 40 | FVACCADR | GENADR | FINDVAC   |                                                       | 40 |
| 41 | WTLTCADR | GENADR | WAITLIST  |                                                       | 41 |
| 42 | NOVACADR | GENADR | NOVAC     |                                                       | 42 |
| 43 |          |        |           |                                                       | 43 |
| 44 |          |        |           |                                                       | 44 |
| 45 |          |        |           |                                                       | 45 |
| 46 |          |        |           |                                                       | 46 |
| 47 |          |        |           |                                                       | 47 |
| 48 |          |        |           |                                                       | 48 |
| 49 |          |        |           |                                                       | 49 |
| 50 |          |        |           |                                                       | 50 |
| 51 |          |        |           |                                                       | 51 |
| 52 |          |        |           |                                                       | 52 |
| 53 |          |        |           |                                                       | 53 |
| 54 |          |        |           |                                                       | 54 |
| 55 |          |        |           |                                                       | 55 |
| 56 |          |        |           |                                                       | 56 |
| 57 |          |        |           |                                                       | 57 |
| 58 |          |        |           |                                                       | 58 |
| 59 |          |        |           |                                                       | 59 |
| 60 |          |        |           |                                                       | 60 |

1412THE

1



# IMU\_MODE\_SWITCHING\_ROUTINES

|    |                        |        |            |    |
|----|------------------------|--------|------------|----|
| 1  |                        |        |            | 1  |
| 2  |                        | BLOCK  | 02         | 2  |
| 3  |                        | SETLOC | FFTAG3     | 3  |
| 4  |                        | BANK   |            | 4  |
| 5  |                        |        |            | 5  |
| 6  |                        | EBANK= | COMMAND    | 6  |
| 7  |                        |        |            | 7  |
| 8  | # FIXED-FIXED ROUTINES |        |            | 8  |
| 9  |                        |        |            | 9  |
| 10 |                        | COUNT* | \$\$/IMODE | 10 |
| 11 | ZEROICDU               | CAF    | ZERO       | 11 |
| 12 |                        | TS     | CDUX       | 12 |
| 13 |                        | TS     | CDUY       | 13 |
| 14 |                        | TS     | CDUZ       | 14 |
| 15 |                        | TC     | Q          | 15 |
| 16 |                        |        |            | 16 |
| 17 | SPSCODE                | =      | BIT9       | 17 |
| 18 |                        |        |            | 18 |
| 19 |                        |        |            | 19 |
| 20 |                        |        |            | 20 |
| 21 |                        |        |            | 21 |
| 22 |                        |        |            | 22 |
| 23 |                        |        |            | 23 |
| 24 |                        |        |            | 24 |
| 25 |                        |        |            | 25 |
| 26 |                        |        |            | 26 |
| 27 |                        |        |            | 27 |
| 28 |                        |        |            | 28 |
| 29 |                        |        |            | 29 |
| 30 |                        |        |            | 30 |
| 31 |                        |        |            | 31 |
| 32 |                        |        |            | 32 |
| 33 |                        |        |            | 33 |
| 34 |                        |        |            | 34 |
| 35 |                        |        |            | 35 |
| 36 |                        |        |            | 36 |
| 37 |                        |        |            | 37 |
| 38 |                        |        |            | 38 |
| 39 |                        |        |            | 39 |
| 40 |                        |        |            | 40 |
| 41 |                        |        |            | 41 |
| 42 |                        |        |            | 42 |
| 43 |                        |        |            | 43 |
| 44 |                        |        |            | 44 |
| 45 |                        |        |            | 45 |
| 46 |                        |        |            | 46 |
| 47 |                        |        |            | 47 |
| 48 |                        |        |            | 48 |
| 49 |                        |        |            | 49 |
| 50 |                        |        |            | 50 |
| 51 |                        |        |            | 51 |
| 52 |                        |        |            | 52 |
| 53 |                        |        |            | 53 |
| 54 |                        |        |            | 54 |
| 55 |                        |        |            | 55 |
| 56 |                        |        |            | 56 |
| 57 |                        |        |            | 57 |
| 58 |                        |        |            | 58 |
| 59 |                        |        |            | 59 |
| 60 |                        |        |            | 60 |

# ZERO ICDU COUNTERS.

## # IMU ZEROING ROUTINES

BANK 11  
SETLOC MODESW  
BANK

IMUZERO

COUNT\* \$\$/IMODE  
INHINT

# ROUTINE TO ZERO ICDUS.

CS DSPTAB +11D  
MASK BITS4&6  
CCS A

# DON'T ZERO CDUS IF IMU IN GIMBAL LOCK AND  
# COARSE ALIGN (GIMBAL RUNAWAY PROTECTION)

TCF IMUZEROA

TC ALARM

# IF SO.

OCT 00206

TCF CAGETSTJ +4

# IMMEDIATE FAILURE.

IMUZEROA

TC CAGETSTJ

CS IMODES33  
MASK SUPER011  
ADS IMODES33

# DISABLE DAP AUTO AND HOLD MODES  
# BIT5 FOR GROUND

CS IMODES30  
MASK BITS3&4  
ADS IMODES30

# INHIBIT ICDUFAIL AND IMUFAIL (IN CASE WE  
# JUST CAME OUT OF COARSE ALIGN).

CS BITS4&6  
EXTEND  
WAND CHAN12

# SEND ZERO ENCODE WITH COARSE AND ERROR  
# COUNTER DISABLED.

TC NOATTOFF

# TURN OFF NO ATT LAMP.

CAF BIT5  
EXTEND  
WOR CHAN12

TC ZEROICDU  
CAF BIT6  
TC WAITLIST  
EBANK= CDUIND  
2CADR IMUZERO2

# WAIT 320 MS TO GIVE AGS ADEQUATE TIME TO  
# RECEIVE ITS PULSE TRAIN.

CS IMODES30  
MASK BIT9  
CCS A

# SEE IF IMU OPERATING AND ALARM IF NOT.

TCF MODEEXIT

|          |                  |                                  |                                                        |
|----------|------------------|----------------------------------|--------------------------------------------------------|
|          | TC<br>OCT        | ALARM<br>210                     |                                                        |
| MODEEXIT | RELINT<br>TCF    | SWRETURN                         | # GENERAL MODE-SWITCHING EXIT.                         |
| IMUZERO2 | TC<br>TC         | CAGETEST<br>ZEROICDU             | # ZERO CDUX, CDUY, CDUZ                                |
|          | CS<br>EXTEND     | BIT5                             | # REMOVE ZERO DISCRETE.                                |
|          | WAND             | CHAN12                           |                                                        |
|          | CAF<br>TC        | BIT11<br>VARDELAY                | # WAIT 10 SECS FOR CTRS TO FIND GIMBALS                |
| IMUZERO3 | TC               | CAGETEST                         |                                                        |
|          | CS<br>MASK<br>TS | BITS3&4<br>IMODES30<br>IMODES30  | # REMOVE IMUFAIL AND ICDUFAIL INHIBIT.                 |
|          | CS<br>MASK<br>TS | SUPER011<br>IMODES33<br>IMODES33 | # ENABLE DAP AUTO AND HOLD MODES<br># BIT5 FOR GROUND  |
|          | TC<br>CADR       | IBNKCALL<br>SETISSW              | # SET ISS WARNING IF EITHER OF ABOVE ARE<br># PRESENT. |
|          | TCF              | ENDIMU                           |                                                        |



# IMU COARSE ALIGN MODE.

IMUCOARS

INHINT

TC

CAGETSTJ

TC

SETCOARS

CAF

SIX

TC

WAITLIST

EBANK=

CDUIND

2CADR

COARS

TCF

MODEEXIT

COARS

TC

CAGETEST

CAF

BIT6

# ENABLE ALL THREE ISS CDU ERROR COUNTERS

EXTEND

WOR

CHAN12

COARS1

CAF

TWO

# SET CDU INDICATOR

TS

CDUIND

INDEX

CDUIND

# COMPUTE THETAD -- THETAA IN 1'S

CA

THETAD

# COMPLEMENT FORM

EXTEND

INDEX

CDUIND

MSU

CDUX

EXTEND

MP

BIT13

# SHIFT RIGHT 2

XCH

L

# ROUND

DOUBLE

TS

ITEMP1

TCF

+2

ADS

L

INDEX

CDUIND

# DIFFERENCE TO BE COMPUTED

LXCH

COMMAND

CCS

CDUIND

TC

COARS1

CAF

TWO

# MINIMUM OF 4 MS WAIT

TC

VARDELAY

|         |        |            |                                           |
|---------|--------|------------|-------------------------------------------|
| COARS2  | TC     | CAGETEST   | # DON'T CONTINUE IF CAGED.                |
|         | TS     | ITEMP1     | # SET TO +0.                              |
| +3      | CAF    | TWO        | # SET CDU INDICATOR.                      |
|         | TS     | CDUIND     |                                           |
|         | INDEX  | CDUIND     |                                           |
|         | CCS    | COMMAND    | # NUMBER OF PULSES REQUIRED               |
|         | TC     | COMPOS     | # GREATER THAN MAX ALLOWED                |
|         | TC     | NEXTCDU +1 |                                           |
|         | TC     | COMNEG     |                                           |
|         | TC     | NEXTCDU +1 |                                           |
| COMPOS  | AD     | -COMMAX    | # COMMAX = MAX NUMBER OF PULSES ALLOWED   |
|         | EXTEND |            | # MINUS ONE                               |
|         | BZMF   | COMZERO    |                                           |
|         | INDEX  | CDUIND     |                                           |
|         | TS     | COMMAND    | # REDUCE COMMAND BY MAX NUMBER OF PULSES  |
|         | CS     | -COMMAX-   | # ALLOWED                                 |
| NEXTCDU | INCR   | ITEMP1     |                                           |
|         | AD     | NEGO       |                                           |
|         | INDEX  | CDUIND     |                                           |
|         | TS     | CDUXCMD    | # SET UP COMMAND REGISTER.                |
|         | CCS    | CDUIND     |                                           |
|         | TC     | COARS2 +3  |                                           |
|         | CCS    | ITEMP1     | # SEE IF ANY PULSES TO GO OUT.            |
|         | TCF    | SENDPULS   |                                           |
|         | TC     | FIXDELAY   | # WAIT FOR GIMBALS TO SETTLE.             |
|         | DEC    | 150        |                                           |
| CHKCORS | CAF    | TWO        | # AT END OF COMMAND, CHECK TO SEE THAT    |
|         | TS     | ITEMP1     | # GIMBALS ARE WITHIN 2 DEGREES OF THETAD. |
|         | INDEX  | A          |                                           |
|         | CA     | CDUX       |                                           |
|         | EXTEND |            |                                           |
|         | INDEX  | ITEMP1     |                                           |
|         | MSU    | THETAD     |                                           |
|         | CCS    | A          |                                           |
|         | TCF    | COARSERR   |                                           |
|         | TCF    | CORSCHK2   |                                           |
|         | TCF    | COARSERR   |                                           |

|    |          |        |           |                                           |    |
|----|----------|--------|-----------|-------------------------------------------|----|
| 1  |          |        |           |                                           | 1  |
| 2  | CORSCHK2 | CCS    | ITEMP1    |                                           | 2  |
| 3  |          | TCF    | CHKCORS   |                                           | 3  |
| 4  |          | TCF    | ENDIMU    | # END OF COARSE ALIGNMENT                 | 4  |
| 5  |          |        |           |                                           | 5  |
| 6  | COARSERR | AD     | COARSTOL  | # 2 DEGREES.                              | 6  |
| 7  |          | EXTEND |           |                                           | 7  |
| 8  |          | BZMF   | CORSCHK2  |                                           | 8  |
| 9  |          |        |           |                                           | 9  |
| 10 |          | TC     | ALARM     | # COARSE ALIGN ERROR.                     | 10 |
| 11 |          | OCT    | 211       |                                           | 11 |
| 12 |          |        |           |                                           | 12 |
| 13 |          | TCF    | IMUBAD    |                                           | 13 |
| 14 |          |        |           |                                           | 14 |
| 15 | COARSTOL | DEC    | -.01111   | # 2 DEGREES SCALED AT HALF-REVOLUTIONS    | 15 |
| 16 |          |        |           |                                           | 16 |
| 17 | COMNEG   | AD     | -COMMAX   |                                           | 17 |
| 18 |          | EXTEND |           |                                           | 18 |
| 19 |          | BZMF   | COMZERO   |                                           | 19 |
| 20 |          | COM    |           |                                           | 20 |
| 21 |          | INDEX  | CDUIND    |                                           | 21 |
| 22 |          | TS     | COMMAND   |                                           | 22 |
| 23 |          | CA     | -COMMAX-  |                                           | 23 |
| 24 |          | TC     | NEXTCDU   |                                           | 24 |
| 25 |          |        |           |                                           | 25 |
| 26 | COMZERO  | CAF    | ZERO      |                                           | 26 |
| 27 |          | INDEX  | CDUIND    |                                           | 27 |
| 28 |          | XCH    | COMMAND   |                                           | 28 |
| 29 |          | TC     | NEXTCDU   |                                           | 29 |
| 30 |          |        |           |                                           | 30 |
| 31 | SENDPULS | CAF    | 13,14,15  |                                           | 31 |
| 32 |          | EXTEND |           |                                           | 32 |
| 33 |          | WOR    | CHAN14    |                                           | 33 |
| 34 |          | CAF    | 600MS     |                                           | 34 |
| 35 |          | TCF    | COARS2 -1 | # THEN TO VARDELAY                        | 35 |
| 36 |          |        |           |                                           | 36 |
| 37 | CA+ECE   | CAF    | BIT6      | # ENABLE ALL THREE ISS CDU ERROR COUNTERS | 37 |
| 38 |          | EXTEND |           |                                           | 38 |
| 39 |          | WOR    | CHAN12    |                                           | 39 |
| 40 |          | TC     | TASKOVER  |                                           | 40 |
| 41 |          |        |           |                                           | 41 |
| 42 |          |        |           |                                           | 42 |
| 43 |          |        |           |                                           | 43 |
| 44 |          |        |           |                                           | 44 |
| 45 |          |        |           |                                           | 45 |
| 46 |          |        |           |                                           | 46 |
| 47 |          |        |           |                                           | 47 |
| 48 |          |        |           |                                           | 48 |
| 49 |          |        |           |                                           | 49 |
| 50 |          |        |           |                                           | 50 |
| 51 |          |        |           |                                           | 51 |
| 52 |          |        |           |                                           | 52 |
| 53 |          |        |           |                                           | 53 |
| 54 |          |        |           |                                           | 54 |
| 55 |          |        |           |                                           | 55 |
| 56 |          |        |           |                                           | 56 |
| 57 |          |        |           |                                           | 57 |
| 58 |          |        |           |                                           | 58 |
| 59 |          |        |           |                                           | 59 |
| 60 |          |        |           |                                           | 60 |

|          |                   |                                        |                                     |
|----------|-------------------|----------------------------------------|-------------------------------------|
| SETCOARS | CAF<br>EXTEND     | BIT4                                   | # BYPASS IF ALREADY IN COARSE ALIGN |
|          | RAND<br>CCS<br>TC | CHAN12<br>A<br>Q                       |                                     |
|          | CS<br>EXTEND      | BIT6                                   | # CLEAR ISS ERROR COUNTERS          |
|          | WAND              | CHAN12                                 |                                     |
|          | CS<br>EXTEND      | BIT10                                  | # KNOCK DOWN GYRO ACTIVITY          |
|          | WAND<br>CS<br>TS  | CHAN14<br>ZERO<br>GYROCMD              |                                     |
|          | CAF<br>EXTEND     | BIT4                                   | # PUT ISS IN COARSE ALIGN           |
|          | WOR               | CHAN12                                 |                                     |
|          | CS<br>MASK<br>ADS | DSPTAB +11D<br>OCT40010<br>DSPTAB +11D | # TURN ON NO ATT LAMP               |
|          | CS<br>MASK<br>ADS | IMODES33<br>BIT6<br>IMODES33           | # DISABLE DAP AUTO AND HOLD MODES   |
|          | CS<br>MASK<br>ADS | IMODES30<br>BIT4<br>IMODES30           | # DISABLE IMUFAIL                   |
| RNDREFDR | CS<br>MASK<br>TS  | TRACKBIT<br>FLAGWRD1<br>FLAGWRD1       | # CLEAR TRACK FLAG                  |
|          | CS<br>MASK<br>TS  | DRFTBIT<br>FLAGWRD2<br>FLAGWRD2        | # CLEAR DRIFT FLAG                  |
|          | CS<br>MASK<br>TS  | REFSMBIT<br>FLAGWRD3<br>FLAGWRD3       | # CLEAR REFSMMAT FLAG               |
|          | TC                | Q                                      |                                     |
| OCT40010 | OCT               | 40010                                  |                                     |

# IMU\_MODE\_SWITCHING\_ROUTINES

# IMU FINE ALIGN MODE SWITCH.

|          |                              |                                         |                                                                                   |
|----------|------------------------------|-----------------------------------------|-----------------------------------------------------------------------------------|
| IMUFINE  | INHINT<br>TC                 | CAGETSTJ                                | # SEE IF IMU BEING CAGED.                                                         |
|          | CS<br>EXTEND<br>WAND         | BITS4-5<br>CHAN12                       | # RESET ZERO AND COARSE                                                           |
|          | CS<br>MASK<br>TS             | BIT6<br>IMODES33<br>IMODES33            | # INSURE DAP AUTO AND HOLD MODES ENABLED                                          |
|          | TC                           | NOATTOFF                                |                                                                                   |
|          | CAF<br>TC                    | BIT10<br>WAITLIST                       | # IMU FAIL WAS INHIBITED DURING THE<br># PRESUMABLY PRECEDING COARSE ALIGN. LEAVE |
|          | EBANK=<br>2CADR              | CDUIND<br>IFAILOK                       | # IT ON FOR THE FIRST 5 SECS OF FINE ALIGN                                        |
|          | CAF<br>TC<br>EBANK=<br>2CADR | 2SECS<br>WAITLIST<br>CDUIND<br>IMUFINED |                                                                                   |
|          | TCF                          | MODEEXIT                                |                                                                                   |
| IMUFINED | TC<br>TCF                    | CAGETEST<br>ENDIMU                      | # SEE THAT NO ONE HAS CAGED THE IMU.                                              |

IFAILOK

TC  
TCFCAGETSTQ  
TASKOVER# ENABLE IMU FAIL UNLESS IMU BEING CAGED.  
# IT IS.CAF  
EXTEND

BIT4

# DON'T RESET IMU FAIL INHIBIT IF SOMEONE  
# HAS GONE INTO COARSE ALIGN.RAND  
CCS  
TCFCHAN12  
A  
TASKOVERCS  
MASKIMODES30  
BIT13

# RESET IMUFAIL.

ADS  
CS  
MASKIMODES30  
BIT4  
IMODES30

PFAILOK2

TS  
TC  
CADRIMODES30  
IBNKCALL  
SETISSW# THE ISS WARNING LIGHT MAY COME ON NOW  
# THAT THE INHIBIT WAS BEEN REMOVED.

TCF

TASKOVER

PFAILOK

TC  
TCFCAGETSTQ  
TASKOVER

# ENABLE PIP FAIL PROG ALARM.

CS  
MASKIMODES30  
BIT10

# RESET IMU AND PIPA FAIL BITS.

ADS

IMODES30

CS  
MASK  
ADSIMODES33  
BIT13  
IMODES33CS  
TCFBIT5  
PFAILOK2

NOATTOFF

CS  
MASKOCT40010  
DSPTAB +11D

# SUBROUTINE TO TURN OFF NO ATT LAMP.

AD  
TS  
TCBIT15  
DSPTAB +11D  
Q

# ROUTINES TO INITIATE AND TERMINATE PROGRAM USE OF THE PIPAS. NO IMUSTALL REQUIRED IN EITHER CASE.

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| PIPUSE   | CS     | ZERO     |                                           |
|          | TS     | PIPAX    |                                           |
|          | TS     | PIPAY    |                                           |
|          | TS     | PIPAZ    |                                           |
| PIPUSE1  | TC     | CAGETSTQ | # DO NOT ENABLE PIPA FAIL IF IMU IS CAGED |
|          | TCF    | SWRETURN |                                           |
|          | INHINT |          |                                           |
|          | CS     | BIT1     | # IF PIPA FAILS FROM NOW ON (UNTIL        |
|          | MASK   | IMODES30 | # PIPFREE), LIGHT ISS WARNING.            |
|          | TS     | IMODES30 |                                           |
| PIPFREE2 | TC     | IBNKCALL | # ISS WARNING MIGHT COME ON NOW.          |
|          | CADR   | SETISSW  | # (OR GO OFF ON PIPFREE).                 |
|          | TCF    | MODEEXIT |                                           |
| PIPFREE  | INHINT |          | # PROGRAM DONE WITH PIPAS. DON'T LIGHT    |
|          | CS     | IMODES30 | # ISS WARNING.                            |
|          | MASK   | BIT1     |                                           |
|          | ADS    | IMODES30 |                                           |
|          | MASK   | BIT10    | # IF PIP FAIL ON, DO PROG ALSRM AND RESET |
|          | CCS    | A        | # ISS WARNING.                            |
|          | TCF    | MODEEXIT |                                           |
|          | TC     | ALARM    |                                           |
|          | OCT    | 212      |                                           |
|          | INHINT |          |                                           |
|          | TCF    | PIPFREE2 |                                           |

```
THE FOLLOWING ROUTINE TORQUES THE IRIGS ACCORDING TO DOUBLE PRECISION INPUTS IN THE SIX REGISTERS
BEGINNING AT THE ECADR ARRIVING IN A. THE MINIMUM SIZE OF ANY PULSE TRAIN IS 16 PULSES (.25 CDU COUNTS). THE
UNSENT PORTION OF THE COMMAND IS LEFT INTACT IN THE INPUT COMMAND REGISTERS.
```

```
EBANK= 1400 # VARIABLE, ACTUALLY.
```

```
IMUPULSE TS MPAC +5 # SAVE ARRIVING ECADR.
TC CAGETSTJ # DON'T PROCEED IF IMU BEING CAGED.
```

```
CCS LGYRO # SEE IF GYROS BUSY.
TC GYROBUSY # SLEEP.
```

```
TS MPAC +2
CAF BIT6 # ENABLE THE POWER SUPPLY.
```

```
EXTEND
WOR CHAN14
```

```
GWAKE2 CAF FOUR
TC WAITLIST # (IF A JOB WAS PUT TO SLEEP, THE POWER
EBANK= CDUIND # SUPPLY IS LEFT ON BY THE WAKING JOB).
2CADR STRTGYRO
```

```
CA MPAC +5 # SET UP EBANK, SAVING CALLER'S EBANK FOR
XCH EBANK # RESTORATION ON RETURN.
```

```
XCH MPAC +5
TS LGYRO # RESERVES GYROS.
```

```
MASK LOW8
TS ITEMP1
```

```
GYROAGRE CAF TWO # FORCE SIGN AGREEMENT ON INPUTS.
TS MPAC +3
DOUBLE
```

```
AD ITEMP1
TS MPAC +4
EXTEND
```

```
INDEX A
DCA 1400
DXCH MPAC
```

```
TC TPAGREE
DXCH MPAC
INDEX MPAC +4
DXCH 1400
```

```
CCS MPAC +3
TCF GYROAGRE
```

```
CA MPAC +5 # RESTORE CALLER'S EBANK.
TS EBANK
TCF MODEEXIT
```





|    |                                                      |        |                       |                                           |    |
|----|------------------------------------------------------|--------|-----------------------|-------------------------------------------|----|
| 1  | # ROUTINES TO ALLOW TORQUING ONLY ONE JOB AT A TIME. |        |                       |                                           | 1  |
| 2  |                                                      |        |                       |                                           | 2  |
| 3  |                                                      |        |                       |                                           | 3  |
| 4  | GYROBUSY                                             | EXTEND | # SAVE RETURN 2FCADR. |                                           | 4  |
| 5  |                                                      | DCA    | BUF2                  |                                           | 5  |
| 6  |                                                      | DXCH   | MPAC                  |                                           | 6  |
| 7  | REGSLEEP                                             | CAF    | LGWAKE                |                                           | 7  |
| 8  |                                                      | TCF    | JOBSLEEP              |                                           | 8  |
| 9  |                                                      |        |                       |                                           | 9  |
| 10 | GWAKE                                                | CCS    | LGYRO                 | # WHEN AWAKENED, SEE IF GYROS STILL BUSY. | 10 |
| 11 |                                                      | TCF    | REGSLEEP              | # IF SO, SLEEP SOME MORE.                 | 11 |
| 12 |                                                      |        |                       |                                           | 12 |
| 13 |                                                      | TS     | MPAC +2               |                                           | 13 |
| 14 |                                                      | EXTEND |                       |                                           | 14 |
| 15 |                                                      | DCA    | MPAC                  |                                           | 15 |
| 16 |                                                      | DXCH   | BUF2                  | # RESTORE SWRETURN INFO.                  | 16 |
| 17 |                                                      | CAF    | ONE                   |                                           | 17 |
| 18 |                                                      | TCF    | GWAKE2                |                                           | 18 |
| 19 |                                                      |        |                       |                                           | 19 |
| 20 | LGWAKE                                               | CADR   | GWAKE                 |                                           | 20 |
| 21 |                                                      |        |                       |                                           | 21 |
| 22 |                                                      |        |                       |                                           | 22 |
| 23 |                                                      |        |                       |                                           | 23 |
| 24 |                                                      |        |                       |                                           | 24 |
| 25 |                                                      |        |                       |                                           | 25 |
| 26 |                                                      |        |                       |                                           | 26 |
| 27 |                                                      |        |                       |                                           | 27 |
| 28 |                                                      |        |                       |                                           | 28 |
| 29 |                                                      |        |                       |                                           | 29 |
| 30 |                                                      |        |                       |                                           | 30 |
| 31 |                                                      |        |                       |                                           | 31 |
| 32 |                                                      |        |                       |                                           | 32 |
| 33 |                                                      |        |                       |                                           | 33 |
| 34 |                                                      |        |                       |                                           | 34 |
| 35 |                                                      |        |                       |                                           | 35 |
| 36 |                                                      |        |                       |                                           | 36 |
| 37 |                                                      |        |                       |                                           | 37 |
| 38 |                                                      |        |                       |                                           | 38 |
| 39 |                                                      |        |                       |                                           | 39 |
| 40 |                                                      |        |                       |                                           | 40 |
| 41 |                                                      |        |                       |                                           | 41 |
| 42 |                                                      |        |                       |                                           | 42 |
| 43 |                                                      |        |                       |                                           | 43 |
| 44 |                                                      |        |                       |                                           | 44 |
| 45 |                                                      |        |                       |                                           | 45 |
| 46 |                                                      |        |                       |                                           | 46 |
| 47 |                                                      |        |                       |                                           | 47 |
| 48 |                                                      |        |                       |                                           | 48 |
| 49 |                                                      |        |                       |                                           | 49 |
| 50 |                                                      |        |                       |                                           | 50 |
| 51 |                                                      |        |                       |                                           | 51 |
| 52 |                                                      |        |                       |                                           | 52 |
| 53 |                                                      |        |                       |                                           | 53 |
| 54 |                                                      |        |                       |                                           | 54 |
| 55 |                                                      |        |                       |                                           | 55 |
| 56 |                                                      |        |                       |                                           | 56 |
| 57 |                                                      |        |                       |                                           | 57 |
| 58 |                                                      |        |                       |                                           | 58 |
| 59 |                                                      |        |                       |                                           | 59 |
| 60 |                                                      |        |                       |                                           | 60 |

# IMU\_MODE\_SWITCHING\_ROUTINES

# GYRO-TORQUING WAITLIST TASKS.

|          |                      |                            |                                        |
|----------|----------------------|----------------------------|----------------------------------------|
| STRTGYRO | CS<br>EXTEND<br>WAND | GDESELCT<br>CHAN14         | # DE-SELECT LAST GYRO.                 |
|          | TC                   | CAGETEST                   |                                        |
| STRTGYR2 | CA<br>EXTEND<br>MP   | LGYRO<br>BIT4              | # JUMP ON PHASE COUNTER IN BITS 13-14. |
|          | INDEX<br>TCF         | A<br>+1                    |                                        |
|          | TC                   | GSELECT                    | # =0. DO Y GYRO.                       |
|          | OCT                  | 00202                      |                                        |
|          | TC                   | GSELECT                    | # =1. DO Z GYRO.                       |
|          | OCT                  | 00302                      |                                        |
|          | TC                   | GSELECT -2                 | # =2. DO X GYRO.                       |
|          | OCT                  | 00100                      |                                        |
|          | CAF                  | ZERO                       | # =3. DONE                             |
|          | TS<br>CAF<br>TC      | LGYRO<br>LGWAKE<br>JOBWAKE | # WAKE A POSSIBLE SLEEPING JOB.        |
| NORESET  | TCF                  | IMUFINED                   | # DO NOT RESET POWER SUPPLY            |

# IMU\_MODE\_SWITCHING\_ROUTINES

|    |         |        |          |                                            |    |
|----|---------|--------|----------|--------------------------------------------|----|
| 1  |         |        |          |                                            | 1  |
| 2  | -2      | CS     | FOUR     | # SPECIAL ENTRY TO REGRESS LGYRO FOR X.    | 2  |
| 3  |         | ADS    | LGYRO    |                                            | 3  |
| 4  |         |        |          |                                            | 4  |
| 5  | GSELECT | INDEX  | Q        | # SELECT GYRO.                             | 5  |
| 6  |         | CAF    | 0        | # PACKED WORD CONTAINS GYRO SELECT BITS    | 6  |
| 7  |         | TS     | ITEMP4   | # AND INCREMENT TO LGYRO.                  | 7  |
| 8  |         | MASK   | SEVEN    |                                            | 8  |
| 9  |         | AD     | BIT13    |                                            | 9  |
| 10 |         | ADS    | LGYRO    |                                            | 10 |
| 11 |         | TS     | EBANK    |                                            | 11 |
| 12 |         | MASK   | LOW8     |                                            | 12 |
| 13 |         | TS     | ITEMP1   |                                            | 13 |
| 14 |         |        |          |                                            | 14 |
| 15 |         | CS     | SEVEN    |                                            | 15 |
| 16 |         | MASK   | ITEMP4   |                                            | 16 |
| 17 |         | TS     | ITEMP4   |                                            | 17 |
| 18 |         |        |          |                                            | 18 |
| 19 |         | EXTEND |          | # MOVE DP COMMAND TO RUPTREGS FOR TESTING. | 19 |
| 20 |         | INDEX  | ITEMP1   |                                            | 20 |
| 21 |         | DCA    | 1400     |                                            | 21 |
| 22 |         | DXCH   | RUPTREG1 |                                            | 22 |
| 23 |         |        |          |                                            | 23 |
| 24 |         | CCS    | RUPTREG1 |                                            | 24 |
| 25 |         | TCF    | MAJ+     |                                            | 25 |
| 26 |         | TCF    | +2       |                                            | 26 |
| 27 |         | TCF    | MAJ-     |                                            | 27 |
| 28 |         |        |          |                                            | 28 |
| 29 |         | CCS    | RUPTREG2 |                                            | 29 |
| 30 |         | TCF    | MIN+     |                                            | 30 |
| 31 |         | TCF    | STRGTYR2 |                                            | 31 |
| 32 |         | TCF    | MIN-     |                                            | 32 |
| 33 |         | TCF    | STRGTYR2 |                                            | 33 |
| 34 |         |        |          |                                            | 34 |
| 35 |         |        |          |                                            | 35 |
| 36 |         |        |          |                                            | 36 |
| 37 |         |        |          |                                            | 37 |
| 38 |         |        |          |                                            | 38 |
| 39 |         |        |          |                                            | 39 |
| 40 |         |        |          |                                            | 40 |
| 41 |         |        |          |                                            | 41 |
| 42 |         |        |          |                                            | 42 |
| 43 |         |        |          |                                            | 43 |
| 44 |         |        |          |                                            | 44 |
| 45 |         |        |          |                                            | 45 |
| 46 |         |        |          |                                            | 46 |
| 47 |         |        |          |                                            | 47 |
| 48 |         |        |          |                                            | 48 |
| 49 |         |        |          |                                            | 49 |
| 50 |         |        |          |                                            | 50 |
| 51 |         |        |          |                                            | 51 |
| 52 |         |        |          |                                            | 52 |
| 53 |         |        |          |                                            | 53 |
| 54 |         |        |          |                                            | 54 |
| 55 |         |        |          |                                            | 55 |
| 56 |         |        |          |                                            | 56 |
| 57 |         |        |          |                                            | 57 |
| 58 |         |        |          |                                            | 58 |
| 59 |         |        |          |                                            | 59 |
| 60 |         |        |          |                                            | 60 |

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  | MIN+     | AD     | -GYROMIN | # SMALL POSITIVE COMMAND. SEE IF AT LEAST  | 2  |
| 3  |          | EXTEND |          | # 16 GYRO PULSES.                          | 3  |
| 4  |          | BZMF   | STRTGYR2 |                                            | 4  |
| 5  |          |        |          |                                            | 5  |
| 6  | MAJ+     | EXTEND |          | # DEFINITE POSITIVE OUTPUT.                | 6  |
| 7  |          | DCA    | GYROFRAC |                                            | 7  |
| 8  |          | DAS    | RUPTREG1 |                                            | 8  |
| 9  |          |        |          |                                            | 9  |
| 10 |          | CA     | ITEMP4   | # SELECT POSITIVE TORQUING FOR THIS GYRO.  | 10 |
| 11 |          | EXTEND |          |                                            | 11 |
| 12 |          | WOR    | CHAN14   |                                            | 12 |
| 13 |          |        |          |                                            | 13 |
| 14 |          | CAF    | LOW7     | # LEAVE NUMBER OF POSSIBLE 8192 AUGMENTS   | 14 |
| 15 |          | MASK   | RUPTREG2 | # TO INITIAL COMMAND IN MAJOR PART OF LONG | 15 |
| 16 |          | XCH    | RUPTREG2 | # TERM STORAGE AND TRUNCATED FRACTION      | 16 |
| 17 | GMERGE   | EXTEND |          | # IN MINOR PART. THE MAJOR PART WILL BE    | 17 |
| 18 |          | MP     | BIT8     | # COUNTED DOWN TO ZERO IN THE COURSE OF    | 18 |
| 19 |          | TS     | ITEMP2   | # PUTTING OUT THE ENTIRE COMMAND.          | 19 |
| 20 |          | CA     | RUPTREG1 |                                            | 20 |
| 21 |          | EXTEND |          |                                            | 21 |
| 22 |          | MP     | BIT9     |                                            | 22 |
| 23 |          | TS     | RUPTREG1 |                                            | 23 |
| 24 |          | CA     | L        |                                            | 24 |
| 25 |          | EXTEND |          |                                            | 25 |
| 26 |          | MP     | BIT14    |                                            | 26 |
| 27 |          | ADS    | ITEMP2   | # INITIAL COMMAND.                         | 27 |
| 28 |          |        |          |                                            | 28 |
| 29 |          | EXTEND |          | # SEE IF MORE THAN ONE PULSE TRAIN NEEDED  | 29 |
| 30 |          | DCA    | RUPTREG1 | # (MORE THAN 16383 PULSES).                | 30 |
| 31 |          | AD     | MINUS1   |                                            | 31 |
| 32 |          | CCS    | A        |                                            | 32 |
| 33 |          | TCF    | LONGGYRO |                                            | 33 |
| 34 | -GYROMIN | OCT    | -176     | # MAY BE ADJUSTED TO SPECIFY MINIMUM CMD   | 34 |
| 35 |          | TCF    | +4       |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 |          | CAF    | BIT14    |                                            | 37 |
| 38 |          | ADS    | ITEMP2   |                                            | 38 |
| 39 |          | CAF    | ZERO     |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 | +4       | INDEX  | ITEMP1   |                                            | 41 |
| 42 |          | DXCH   | 1400     |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          |        |          |                                            | 44 |
| 45 |          |        |          |                                            | 45 |
| 46 |          |        |          |                                            | 46 |
| 47 |          |        |          |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          |        |          |                                            | 49 |
| 50 |          |        |          |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

|    |          |        |            |                                           |
|----|----------|--------|------------|-------------------------------------------|
| 1  |          |        |            |                                           |
| 2  |          | CA     | ITEMP2     | # ENTIRE COMMAND.                         |
| 3  | LASTSEG  | TS     | GYROCMD    |                                           |
| 4  |          | EXTEND |            |                                           |
| 5  |          | MP     | BIT10      | # WAITLIST DT                             |
| 6  |          | AD     | THREE      | # TRUNCATION AND PHASE UNCERTAINTIES.     |
| 7  |          | TC     | WAITLIST   |                                           |
| 8  |          | EBANK= | CDUIND     |                                           |
| 9  |          | 2CADR  | STRGyro    |                                           |
| 10 |          |        |            |                                           |
| 11 | GYROEXIT | CAF    | BIT10      |                                           |
| 12 |          | EXTEND |            |                                           |
| 13 |          | WOR    | CHAN14     |                                           |
| 14 |          | TCF    | TASKOVER   |                                           |
| 15 |          |        |            |                                           |
| 16 | LONGGYRO | INDEX  | ITEMP1     |                                           |
| 17 |          | DXCH   | 1400       | # INITIAL COMMAND OUT PLUS N AUGMENTS OF  |
| 18 |          | CAF    | BIT14      | # 8192. INITIAL COMMAND IS AT LEAST 8192. |
| 19 |          | AD     | ITEMP2     |                                           |
| 20 |          | TS     | GYROCMD    |                                           |
| 21 |          |        |            |                                           |
| 22 | AUG3     | EXTEND |            | # GET WAITLIST DT TO TIME WHEN TRAIN IS   |
| 23 |          | MP     | BIT10      | # ALMOST OUT.                             |
| 24 |          | AD     | NEG3       |                                           |
| 25 |          | TC     | WAITLIST   |                                           |
| 26 |          | EBANK= | CDUIND     |                                           |
| 27 |          | 2CADR  | 8192AUG    |                                           |
| 28 |          |        |            |                                           |
| 29 |          | TCF    | GYROEXIT   |                                           |
| 30 |          |        |            |                                           |
| 31 | 8192AUG  | TC     | CAGETEST   |                                           |
| 32 |          |        |            |                                           |
| 33 |          | CAF    | BIT4       |                                           |
| 34 |          | EXTEND |            |                                           |
| 35 |          | RAND   | CHAN12     |                                           |
| 36 |          | CCS    | A          |                                           |
| 37 |          | TCF    | IMUBAD     |                                           |
| 38 |          | CA     | LGYRO      | # ADD 8192 PULSES TO GYROCMD              |
| 39 |          | TS     | EBANK      |                                           |
| 40 |          | MASK   | LOW8       |                                           |
| 41 |          | TS     | ITEMP1     |                                           |
| 42 |          |        |            |                                           |
| 43 |          | INDEX  | ITEMP1     | # SEE IF THIS IS THE LAST AUG.            |
| 44 |          | CCS    | 1400       |                                           |
| 45 |          | TCF    | AUG2       | # MORE TO COME.                           |
| 46 |          |        |            |                                           |
| 47 |          | CAF    | BIT14      |                                           |
| 48 |          | ADS    | GYROCMD    |                                           |
| 49 |          | TCF    | LASTSEG +1 |                                           |
| 50 |          |        |            |                                           |
| 51 |          |        |            |                                           |
| 52 |          |        |            |                                           |
| 53 |          |        |            |                                           |
| 54 |          |        |            |                                           |
| 55 |          |        |            |                                           |
| 56 |          |        |            |                                           |
| 57 |          |        |            |                                           |
| 58 |          |        |            |                                           |
| 59 |          |        |            |                                           |
| 60 |          |        |            |                                           |



|    |      |       |               |    |
|----|------|-------|---------------|----|
| 1  |      |       |               | 1  |
| 2  | AUG2 | INDEX | ITEMP1        | 2  |
| 3  |      | TS    | 1400          | 3  |
| 4  |      | CAF   | BIT14         | 4  |
| 5  |      | ADS   | GYROCMD       | 5  |
| 6  |      | TCF   | AUG3          | 6  |
| 7  |      |       | # COMPUTE DT. | 7  |
| 8  |      |       |               | 8  |
| 9  |      |       |               | 9  |
| 10 |      |       |               | 10 |
| 11 |      |       |               | 11 |
| 12 |      |       |               | 12 |
| 13 |      |       |               | 13 |
| 14 |      |       |               | 14 |
| 15 |      |       |               | 15 |
| 16 |      |       |               | 16 |
| 17 |      |       |               | 17 |
| 18 |      |       |               | 18 |
| 19 |      |       |               | 19 |
| 20 |      |       |               | 20 |
| 21 |      |       |               | 21 |
| 22 |      |       |               | 22 |
| 23 |      |       |               | 23 |
| 24 |      |       |               | 24 |
| 25 |      |       |               | 25 |
| 26 |      |       |               | 26 |
| 27 |      |       |               | 27 |
| 28 |      |       |               | 28 |
| 29 |      |       |               | 29 |
| 30 |      |       |               | 30 |
| 31 |      |       |               | 31 |
| 32 |      |       |               | 32 |
| 33 |      |       |               | 33 |
| 34 |      |       |               | 34 |
| 35 |      |       |               | 35 |
| 36 |      |       |               | 36 |
| 37 |      |       |               | 37 |
| 38 |      |       |               | 38 |
| 39 |      |       |               | 39 |
| 40 |      |       |               | 40 |
| 41 |      |       |               | 41 |
| 42 |      |       |               | 42 |
| 43 |      |       |               | 43 |
| 44 |      |       |               | 44 |
| 45 |      |       |               | 45 |
| 46 |      |       |               | 46 |
| 47 |      |       |               | 47 |
| 48 |      |       |               | 48 |
| 49 |      |       |               | 49 |
| 50 |      |       |               | 50 |
| 51 |      |       |               | 51 |
| 52 |      |       |               | 52 |
| 53 |      |       |               | 53 |
| 54 |      |       |               | 54 |
| 55 |      |       |               | 55 |
| 56 |      |       |               | 56 |
| 57 |      |       |               | 57 |
| 58 |      |       |               | 58 |
| 59 |      |       |               | 59 |
| 60 |      |       |               | 60 |

# IMU\_MODE\_SWITCHING\_ROUTINES

|    |          |        |            |                                           |    |
|----|----------|--------|------------|-------------------------------------------|----|
| 1  |          |        |            |                                           | 1  |
| 2  | MIN-     | AD     | -GYROMIN   | # POSSIBLE NEGATIVE OUTPUT.               | 2  |
| 3  |          | EXTEND |            |                                           | 3  |
| 4  |          | BZMF   | STRTGYR2   |                                           | 4  |
| 5  |          |        |            |                                           | 5  |
| 6  | MAJ-     | EXTEND |            | # DEFINITE NEGATIVE OUTPUT.               | 6  |
| 7  |          | DCS    | GYROFRAC   |                                           | 7  |
| 8  |          | DAS    | RUPTREG1   |                                           | 8  |
| 9  |          |        |            |                                           | 9  |
| 10 |          | CA     | ITEMP4     | # SELECT NEGATIVE TORQUING FOR THIS GYRO. | 10 |
| 11 |          | AD     | BIT9       |                                           | 11 |
| 12 |          | EXTEND |            |                                           | 12 |
| 13 |          | WOR    | CHAN14     |                                           | 13 |
| 14 |          |        |            |                                           | 14 |
| 15 |          | CS     | RUPTREG1   | # SET UP RUPTREGS TO FALL INTO GMERGE.    | 15 |
| 16 |          | TS     | RUPTREG1   | # ALL NUMBERS PUT INTO GYROCMD ARE        | 16 |
| 17 |          | CS     | RUPTREG2   | # POSITIVE -- BIT9 OF CHAN 14 DETERMINES  | 17 |
| 18 |          | MASK   | LOW7       | # THE SIGN OF THE COMMAND.                | 18 |
| 19 |          |        |            |                                           | 19 |
| 20 |          | COM    |            |                                           | 20 |
| 21 |          | XCH    | RUPTREG2   |                                           | 21 |
| 22 |          | COM    |            |                                           | 22 |
| 23 |          | TCF    | GMERGE     |                                           | 23 |
| 24 | GDESELCT | OCT    | 1700       | # TURN OFF SELECT AND ACTIVITY BITS.      | 24 |
| 25 |          |        |            |                                           | 25 |
| 26 | GYROFRAC | 2DEC   | .215 B -21 |                                           | 26 |
| 27 |          |        |            |                                           | 27 |
| 28 |          |        |            |                                           | 28 |
| 29 |          |        |            |                                           | 29 |
| 30 |          |        |            |                                           | 30 |
| 31 |          |        |            |                                           | 31 |
| 32 |          |        |            |                                           | 32 |
| 33 |          |        |            |                                           | 33 |
| 34 |          |        |            |                                           | 34 |
| 35 |          |        |            |                                           | 35 |
| 36 |          |        |            |                                           | 36 |
| 37 |          |        |            |                                           | 37 |
| 38 |          |        |            |                                           | 38 |
| 39 |          |        |            |                                           | 39 |
| 40 |          |        |            |                                           | 40 |
| 41 |          |        |            |                                           | 41 |
| 42 |          |        |            |                                           | 42 |
| 43 |          |        |            |                                           | 43 |
| 44 |          |        |            |                                           | 44 |
| 45 |          |        |            |                                           | 45 |
| 46 |          |        |            |                                           | 46 |
| 47 |          |        |            |                                           | 47 |
| 48 |          |        |            |                                           | 48 |
| 49 |          |        |            |                                           | 49 |
| 50 |          |        |            |                                           | 50 |
| 51 |          |        |            |                                           | 51 |
| 52 |          |        |            |                                           | 52 |
| 53 |          |        |            |                                           | 53 |
| 54 |          |        |            |                                           | 54 |
| 55 |          |        |            |                                           | 55 |
| 56 |          |        |            |                                           | 56 |
| 57 |          |        |            |                                           | 57 |
| 58 |          |        |            |                                           | 58 |
| 59 |          |        |            |                                           | 59 |
| 60 |          |        |            |                                           | 60 |

## # IMU\_MODE\_SWITCHING\_ROUTINES

# IMU MODE SWITCHING ROUTINES COME HERE WHEN ACTION COMPLETE.

|        |        |          |                                          |
|--------|--------|----------|------------------------------------------|
| ENDIMU | EXTEND |          | # MODE IS BAD IF CAGE HAS OCCURRED OR IF |
|        | READ   | DSALMOUT | # ISS WARNING IS ON.                     |
|        | MASK   | BIT1     |                                          |

|     |        |
|-----|--------|
| CCS | A      |
| TCF | IMUBAD |

|         |     |         |                  |
|---------|-----|---------|------------------|
| IMUGOOD | TCF | GOODEND | # WITH C(A) = 0. |
|---------|-----|---------|------------------|

|        |     |        |
|--------|-----|--------|
| IMUBAD | CAF | ZERO   |
|        | TCF | BADEND |

|          |      |          |                                    |
|----------|------|----------|------------------------------------|
| CAGETEST | CAF  | BIT6     | # SUBROUTINE TO TERMINATE IMU MODE |
|          | MASK | IMODES30 | # SWITCH IF IMU HAS BEEN CAGED.    |

|     |        |
|-----|--------|
| CCS | A      |
| TCF | IMUBAD |

|    |   |                   |
|----|---|-------------------|
| TC | Q | # WITH C(A) = +0. |
|----|---|-------------------|

|          |    |          |                                |
|----------|----|----------|--------------------------------|
| CAGETSTQ | CS | IMODES30 | # SKIP IF IMU NOT BEING CAGED. |
|----------|----|----------|--------------------------------|

|      |      |
|------|------|
| MASK | BIT6 |
| CCS  | A    |
| INCR | Q    |

|    |   |
|----|---|
| TC | Q |
|----|---|

|          |      |          |                                            |
|----------|------|----------|--------------------------------------------|
| CAGETSTJ | CS   | IMODES30 | # IF DURING MODE SWITCH INITIALIZATION     |
|          | MASK | BIT6     | # IT IS FOUND THAT THE IMU IS BEING CAGED, |
|          | CCS  | A        | # SET IMUCADR TO -0 TO INDICATE OPERATION  |
|          | TC   | Q        | # COMPLETE BUT FAILED. RETURN IMMEDIATELY  |

|     |          |                |
|-----|----------|----------------|
| CS  | ZERO     | # TO SWRETURN. |
| TS  | IMUCADR  |                |
| TCF | MODEEXIT |                |



# GENERALIZED MODE SWITCHING TERMINATION. ENTER AT GOODEND FOR SUCCESSFUL COMPLETION OF AN I/O OPERATION  
# OR AT BADEND FOR A N UNSUCCESSFUL ONE. C(A) OR ARRIVAL =0 FOR IMU, 1 FOR OPTICS.

|            |        |            |                                           |
|------------|--------|------------|-------------------------------------------|
| BADEND     | TS     | RUPTREG2   | # DEVICE INDEX.                           |
|            | CS     | ZERO       | # FOR FAILURE.                            |
|            | TCF    | GOODEND +2 |                                           |
| GOODEND    | TS     | RUPTREG2   |                                           |
|            | CS     | ONE        | # FOR SUCCESS.                            |
|            | TS     | RUPTREG3   |                                           |
|            | INDEX  | RUPTREG2   | # SEE IF USING PROGRAM ASLEEP.            |
|            | CCS    | MODECADR   |                                           |
|            | TCF    | +4         | # YES -- WAKE IT UP.                      |
|            | TCF    | ENDMODE    | # IF 0, PROGRAM NOT IN YET.               |
|            | EXTEND |            |                                           |
|            | BZF    | ENDMODE +1 | # BZF = TCF IF MODECADR = -0.             |
|            | CAF    | ZERO       | # WAKE SLEEPING PROGRAM.                  |
|            | INDEX  | RUPTREG2   |                                           |
|            | XCH    | MODECADR   |                                           |
|            | TC     | JOBWAKE    |                                           |
|            | CS     | RUPTREG3   | # ADVANCE LOC IF SUCCESSFUL.              |
|            | INDEX  | LOCCTR     |                                           |
|            | ADS    | LOC        |                                           |
|            | TCF    | TASKOVER   |                                           |
| ENDMODE +1 | CA     | RUPTREG3   | # -0 INDICATES OPERATION COMPLETE BUT     |
|            | INDEX  | RUPTREG2   | # UNSUCCESSFUL: -1 INDICATES COMPLETE AND |
|            | TS     | MODECADR   | # SUCCESSFUL.                             |
|            | TCF    | TASKOVER   |                                           |

# GENERAL STALLING ROUTINE. USING PROGRAMS COME HERE TO WAIT FOR I/O COMPLETION.

# PROGRAM DESCRIPTION

DATE- 21 FEB 1967

LOG SECTION IMU MODE SWITCHING

# MOD BY- R.MELANSON TO ADD DOCUMENTATION

ASSEMBLY SUNDISK REV. 82

# FUNCTIONAL DESCRIPTION-

# TO DELAY FURTHER EXECUTION OF THE CALLING ROUTINE UNTIL ITS SELECTED  
# I/O FUNCTION IS COMPLETE.THE FOLLOWING CHECKS ON THE CALLING ROUTINE:S  
# MODECADR ARE MADE AND ACTED UPON.

# 1) +0 INDICATES INCOMPLETE I/O OPERATION.CALLING ROUTINE IS PUT TO  
# SLEEP.

# 2) -1 INDICATES COMPLETED I/O OPERATION. STALL BYPASSES JOBSLEEP  
# CALL AND RETURNS TO CALLING ROUTINE AT L+3

# 3) -0 INDICATES COMPLETED I/O WITH FAILURE. STALL CLEARS MODECADR  
# AND RETURNS TO CALLING ROUTINE AT L+2.

# 4) VALUE GREATER THAN 0 INDICATES TWO ROUTINES CALLING FOR USE OF  
# SAME DEVICE. STALL EXITS TO ABORT WHICH EXECUTES A PROGRAM  
# RESTART WHICH IN TURN CLEARS ALL MODECADR REGISTERS.

# CALLING SEQUENCE-

# L TC BANKCALL

# L+1 CADR (ONE OF 5 STALL ADDRESSES I.E. IMUSTALL,OPTSTALL,RADSTALL,  
# AOTSTALL,OR ATTSTALL)

# NORMAL-EXIT MODE-

# TCF JOBSLEEP OR TCF MODEEXIT

# ALARM OR ABORT EXIT MODE-

# TC ABORT

# OUTPUT-

# MODECADR= CADR IF JOBSLEEP

# MODECADR=+0 IF I/O COMPLETE

# BUF2=L+3 IF I/O COMPLETE AND GOOD.

# BUF2=L+2 IF I/O COMPLETE BUT FAILED.

# ERASABLE INITIALIZATION-

# BUF2 CONTAINS RETURN ADDRESS PLUS 1,(L+2)

# BUF2+1 CONTAINS FBANK VALUE OF CALLING ROUTINE.

# MODECADR OF CALLING ROUTINE CONTAINS +0,-1,-0 OR CADR RETURN ADDRESS.

# DEBRIS-

# RUPTREG2 AND CALLING ROUTINE MODECADR.

AOTSTALL

CAF  
TC

ONE  
STALL

# AOT.

RADSTALL

CAF  
TCF

TWO  
STALL

|    |          |        |          |                                           |    |
|----|----------|--------|----------|-------------------------------------------|----|
| 1  |          |        |          |                                           | 1  |
| 2  | OPTSTALL | EQUALS | AOTSTALL |                                           | 2  |
| 3  |          |        |          |                                           | 3  |
| 4  | IMUSTALL | CAF    | ZERO     | # IMU.                                    | 4  |
| 5  |          |        |          |                                           | 5  |
| 6  | STALL    | INHINT |          |                                           | 6  |
| 7  |          | TS     | RUPTREG2 | # SAVE DEVICE INDEX.                      | 7  |
| 8  |          | INDEX  | A        | # SEE IF OPERATION COMPLETE.              | 8  |
| 9  |          | CCS    | MODECADR |                                           | 9  |
| 10 |          | TCF    | MODABORT | # ALLOWABLE STATES ARE +0, -1, AND -0.    | 10 |
| 11 |          | TCF    | MODESLP  | # OPERATION INCOMPLETE.                   | 11 |
| 12 |          | TCF    | MODEGOOD | # COMPLETE AND GOOD IF = -1.              | 12 |
| 13 |          |        |          |                                           | 13 |
| 14 | MG2      | INDEX  | RUPTREG2 | # COMPLETE AND FAILED IF -0. RESET TO +0. | 14 |
| 15 |          | TS     | MODECADR | # RETURN TO CALLER.                       | 15 |
| 16 |          | TCF    | MODEEXIT |                                           | 16 |
| 17 |          |        |          |                                           | 17 |
| 18 | MODEGOOD | CCS    | A        | # MAKE SURE INITIAL STATE -1.             | 18 |
| 19 |          | TCF    | MODABORT |                                           | 19 |
| 20 |          |        |          |                                           | 20 |
| 21 |          | INCR   | BUF2     | # IF SO, INCREMENT RETURN ADDRESS AND     | 21 |
| 22 |          | TCF    | MG2      | # RETURN IMMEDIATELY, SETTING CADR = +0.  | 22 |
| 23 |          |        |          |                                           | 23 |
| 24 | MODESLP  | TC     | MAKECADR | # CALL FROM SWITCHABLE FIXED ONLY.        | 24 |
| 25 |          | INDEX  | RUPTREG2 |                                           | 25 |
| 26 |          | TS     | MODECADR |                                           | 26 |
| 27 |          | TCF    | JOBSLEEP |                                           | 27 |
| 28 |          |        |          |                                           | 28 |
| 29 | MODABORT | DXCH   | BUF2     |                                           | 29 |
| 30 |          | TC     | BAILOUT1 | # TWO PROGRAMS USING THE SAME DEVICE.     | 30 |
| 31 |          | OCT    | 1210     |                                           | 31 |
| 32 |          |        |          |                                           | 32 |
| 33 |          |        |          |                                           | 33 |
| 34 |          |        |          |                                           | 34 |
| 35 |          |        |          |                                           | 35 |
| 36 |          |        |          |                                           | 36 |
| 37 |          |        |          |                                           | 37 |
| 38 |          |        |          |                                           | 38 |
| 39 |          |        |          |                                           | 39 |
| 40 |          |        |          |                                           | 40 |
| 41 |          |        |          |                                           | 41 |
| 42 |          |        |          |                                           | 42 |
| 43 |          |        |          |                                           | 43 |
| 44 |          |        |          |                                           | 44 |
| 45 |          |        |          |                                           | 45 |
| 46 |          |        |          |                                           | 46 |
| 47 |          |        |          |                                           | 47 |
| 48 |          |        |          |                                           | 48 |
| 49 |          |        |          |                                           | 49 |
| 50 |          |        |          |                                           | 50 |
| 51 |          |        |          |                                           | 51 |
| 52 |          |        |          |                                           | 52 |
| 53 |          |        |          |                                           | 53 |
| 54 |          |        |          |                                           | 54 |
| 55 |          |        |          |                                           | 55 |
| 56 |          |        |          |                                           | 56 |
| 57 |          |        |          |                                           | 57 |
| 58 |          |        |          |                                           | 58 |
| 59 |          |        |          |                                           | 59 |
| 60 |          |        |          |                                           | 60 |

# CONSTANTS FOR MODE SWITCHING ROUTINES

|          |        |            |                                  |
|----------|--------|------------|----------------------------------|
| BITS3&4  | =      | OCT14      |                                  |
| BITS4&6  | =      | OCT50      |                                  |
| BITS4-5  | OCT    | 00030      |                                  |
| IMUSEFLG | EQUALS | BIT8       | # INTERPRETER SWITCH 7.          |
| -COMMAX  | DEC    | -191       |                                  |
| -COMMAX- | DEC    | -192       |                                  |
| 600MS    | DEC    | 60         |                                  |
| IMUFIN20 | =      | IMUFINE    |                                  |
| GOMANUR  | CA     | ATTCADR    | # IS KALCMANU FREE               |
|          | EXTEND |            |                                  |
|          | BZF    | +3         |                                  |
|          | TC     | POOD00     | # NO                             |
|          | OCT    | 1210       | # 2 TRYING TO USE SAME DEVICE    |
| +3       | EXTEND |            |                                  |
|          | DCA    | BUF2       |                                  |
|          | DXCH   | ATTCADR    | # SAVE FINAL RETURN FOR KALCMAN3 |
|          | CA     | BBANK      |                                  |
|          | MASK   | SEVEN      |                                  |
|          | ADS    | ATTCADR +1 |                                  |
|          | CA     | PRIORITY   |                                  |
|          | MASK   | PRI037     |                                  |
|          | TS     | ATTPRIO    | # SAVE USERS PRIO                |
|          | CAF    | KALEBCON   | # SET EBANK FOR KALCMAN3         |
|          | TS     | EBANK      |                                  |
|          | TC     | POSTJUMP   |                                  |
| KALEBCON | CADR   | KALCMAN3   |                                  |
|          | ECADR  | BCDU       |                                  |

```
1 # PROGRAM DESCRIPTION
2 # IMU STATUS CHECK ROUTINE R02 (SUBROUTINE UTILITY)
3 # MOD NO - 1
4 # MOD BY - N.BRODEUR
5 # FUNCTIONAL DESCRIPTION
6 #
7 # TO CHECK WHETHER IMU IS ON AND IF ON WHETHER IT IS ALIGNED TO AN
8 # ORIENTATION KNOWN BY THE CMC. TO REQUEST SELECTION OF THE APPROPRIATE
9 # PROGRAM IF THE IMU IS OFF OR NOT ALIGNED TO AN ORIENTATION KNOWN BY THE
10 # CMC. CALLED THROUGH BANKCALL
11 # CALLING SEQUENCE-
12 #
13 # L TC BANKCALL
14 # L+1 CADR R02BOTH
15 # SUBROUTINES CALLED
16 #
17 # VARALARM
18 # FLAGUP
19 # NORMAL EXIT MODES
20 #
21 # AT L+2 OF CALLING SEQUENCE
22 # ALARM OR ABORT EXIT MODES
23 # GOTOPPOH, WITH ALARM
24 # ERASABLE INITIALIZATION REQUIRED
25 #
26 # NONE
27 # DEBRIS
28 #
29 # CENTRALS-A,Q,L
```

```
31
32 BANK 34
33 SETLOC R02
34 BANK
35 COUNT* $$/R02
36 DEC51 DEC 51
37 R02BOTH CAF REFSMBIT
38 MASK FLAGWRD3
39 CCS A
40 TC R02ZERO # ZERO IMUS
41
42 CA IMODES30
43 MASK BIT9 # IS ISS INITIALIZED
44 EXTEND
45 BZF +2
46 CS BIT4 # SEND IMU ALARM CODE 210
47 AD OCT220 # SEND REFSMM ALARM
48 TC VARALARM
49
50 TC GOTOPPOH
51
52 R02ZERO TC UPFLAG
```



|    |        |       |          |    |
|----|--------|-------|----------|----|
| 1  |        |       |          | 1  |
| 2  |        |       |          | 2  |
| 3  |        |       |          | 3  |
| 4  |        | ADRES | IMUSE    | 4  |
| 5  |        | TCF   | SWRETURN | 5  |
| 6  | OCT220 | OCT   | 220      | 6  |
| 7  |        |       |          | 7  |
| 8  |        |       |          | 8  |
| 9  |        |       |          | 9  |
| 10 |        |       |          | 10 |
| 11 |        |       |          | 11 |
| 12 |        |       |          | 12 |
| 13 |        |       |          | 13 |
| 14 |        |       |          | 14 |
| 15 |        |       |          | 15 |
| 16 |        |       |          | 16 |
| 17 |        |       |          | 17 |
| 18 |        |       |          | 18 |
| 19 |        |       |          | 19 |
| 20 |        |       |          | 20 |
| 21 |        |       |          | 21 |
| 22 |        |       |          | 22 |
| 23 |        |       |          | 23 |
| 24 |        |       |          | 24 |
| 25 |        |       |          | 25 |
| 26 |        |       |          | 26 |
| 27 |        |       |          | 27 |
| 28 |        |       |          | 28 |
| 29 |        |       |          | 29 |
| 30 |        |       |          | 30 |
| 31 |        |       |          | 31 |
| 32 |        |       |          | 32 |
| 33 |        |       |          | 33 |
| 34 |        |       |          | 34 |
| 35 |        |       |          | 35 |
| 36 |        |       |          | 36 |
| 37 |        |       |          | 37 |
| 38 |        |       |          | 38 |
| 39 |        |       |          | 39 |
| 40 |        |       |          | 40 |
| 41 |        |       |          | 41 |
| 42 |        |       |          | 42 |
| 43 |        |       |          | 43 |
| 44 |        |       |          | 44 |
| 45 |        |       |          | 45 |
| 46 |        |       |          | 46 |
| 47 |        |       |          | 47 |
| 48 |        |       |          | 48 |
| 49 |        |       |          | 49 |
| 50 |        |       |          | 50 |
| 51 |        |       |          | 51 |
| 52 |        |       |          | 52 |
| 53 |        |       |          | 53 |
| 54 |        |       |          | 54 |
| 55 |        |       |          | 55 |
| 56 |        |       |          | 56 |
| 57 |        |       |          | 57 |
| 58 |        |       |          | 58 |
| 59 |        |       |          | 59 |
| 60 |        |       |          | 60 |



# IMU\_MODE\_SWITCHING\_ROUTINES

# PROGRAM DESCRIPTION P06 10FEB67  
#  
# TRANSFER THE ISS/CMC FROM THE OPERATE TO THE STANDBY CONDITION.  
#  
# THE NORMAL CONDITION OF READINESS OF THE GNCS WHEN NOT IN USE IS STANDBY. IN THIS CONDITION THE IMU  
# HEATER POWER IS ON. THE IMU OPERATE POWER IS OFF. THE COMPUTER POWER IS ON. THE OPTICS POWER IS OFF. THE  
# CMC STANDBY ON THE MAIN AND LEB DISKYS IS ON.  
#  
# CALLING SEQUENCE:  
# ASTRONAUT REQUEST THROUGH DSKY V37E 06E.  
#  
# SUBROUTINES CALLED:  
# GOPERF1  
# BANKCALL  
# FLAGDOWN  
#

```
1 # PRESTAND PREPARES FOR STANDBY BY SNAPSHOTTING THE SCALER AND TIME1 TIME2
2 # THE LOW 5 BITS OF THE SCALER ARE INSPECTED TO INSURE COMPATIBILITY
3 # BETWEEN THE SCALER READING AND THE TIME1 TIME2 READING.
4
5
6 SETLOC P05P06
7 BANK
8
9 EBANK= TIME2SAV
10 COUNT* $$/P06
11
12 P06 TC UPFLAG # SET NODOV37 BIT
13 ADRES NODOFLAG
14
15 PRESTAND INHINT
16 EXTEND
17 DCA TIME2 # SNAPSHOT TIME1TIME2
18 DXCH TIME2SAV
19 TC SCALPREP
20 TC PRESTAND # T1,T2,SCALER NOT COMPATIBLE
21 DXCH MPAC # T1,T2 AND SCALER OK
22 DXCH SCALSAVE # STORE SCALER
23 INHINT
24 TC BANKCALL
25 CADR RNDREFDR # REFSMM, DRIFT, TRACK FLAGS DOWN
26
27 TC DOWNFLAG
28 ADRES IMUSE # IMUSE DOWN
29 TC DOWNFLAG
30 ADRES RNDVZFLG # RNDVZFLG DOWN
31
32 CAF BIT11
33 EXTEND
34 WOR CHAN13 # SET STANDBY ENABLE BIT
35
36 TC PHASCHNG # SET RESTART TO POSTAND WHEN STANDBY
37 OCT 07024 # RECOVERS
38 OCT 20000
39 EBANK= SCALSAVE
40 2CADR POSTAND
41
42 CAF OCT62
43 TC BANKCALL
44 CADR GOPERF1
45 TCF -3
46 TCF -4
47 TCF -5
48
49 OCT62 EQUALS .5SEC # DEC 50 = OCT 62
50
51 # THE LOW 5 BITS OF THE SCALER READS 10000 FOR THE FIRST INTERVAL AFTER A
```



```
1 # T1 INCREMENT. IF SCALPREP DETECTS THIS INTERVAL THE T1,T2 AND SCALER
2 # DATA ARE NOT COMPATIBLE AND RETURN IS TO L+1 FOR ANOTHER READING OF THE
3 # DATA. OTHERWISE, THE RETURN IS TO L+2 TO PROCEED. ROUTINE ALSO PREPARES
4 # THE SCALER READING FOR COMPUTATION OF THE INCREMENT TO UPDATE T1T2. (THE
5 # 10 MS BIT (BIT 6) OF THE SCALER IS INCREMENTED 5 MS OUT OF PHASE FROM
6 # T1.) ADDITION OF 5 MS (BIT 5) TO THE SCALER READING HAS THE EFFECT OF
7 # ADJUSTING BIT 6 IN THE SCALER TO BE IN PHASE WITH BIT 1 OF T1. THE LOW 5
8 # BITS OF THE SCALER READING ARE THEN SET TO ZERO, TO TRUNCATE THE SCALER
9 # DATA TO 10 MS. RESULTS ARE STORED IN MPAC, +1.
```

```
11 SCALPREP EXTEND
```

```
12 QXCH MPAC +2
13 TC FINETIME +1
14 RELINT
```

```
15 DXCH MPAC
16 CA BIT5 # ADD 5 MS TO THE SCALER READING.
17 TS L
```

```
18 CA ZERO
19 DAS MPAC
20 CS LOW5 # SET LOW 5 BITS OF (SCALER+5MS) TO ZERO
```

```
21 MASK MPAC +1 # AND STORE RESULTS IN MPAC,+1.
```

```
22 XCH MPAC +1
```

```
23 MASK LOW5
```

```
24 # TEST LOW 5 BITS OF SCALER FOR THE FIRST
```

```
25 # INTERVAL AFTER THE T1 INCREMENT
```

```
26 # (NOW = 00000, SINCE BIT 5 ADDED).
```

```
27 CCS A
```

```
28 # IS IT 1ST INTERVAL AFTER T1 INCREMENT
```

```
29 INCR MPAC +2
```

```
30 # NO
```

```
31 TC MPAC +2
```

```
32 # YES
```

```
33 # POSTAND RECOVERS TIME AFTER STANDBY.THE SCALER IS SNAPSHOTTED AND THE
34 # TIME1 TIME2 COUNTER IS SET TO ZERO. THE LOW 5 BITS OF THE SCALER ARE
35 # INSPECTED TO INSURE COMPATIBILITY BETWEEN THE SCALER READING AND THE
36 # CLEARING OF THE TIME COUNTER. IT THEN COMPUTES THE DIFFERENCE IN SCALER
37 # VALUES (IN DP) AND ADDS THIS TO THE PREVIOUSLY SNAPSHOTTED VALUES OF
38 # TIME1 TIME2 AND PLACES THIS NEW TIME INTO THE TIME1 TIME2 COUNTER.
```

```
39 COUNT* $$/P05
```

```
40 POSTAND CS BIT11 # RECOVER TIME AFTER STANDBY.
```

```
41 EXTEND
```

```
42 WAND CHAN13 # CLEAR STANDBY ENABLE BIT
```

```
43 INHINT
```

```
44 CA ZERO
```

```
45 TS L
```

```
46 DXCH TIME2
```

```
47 # CLEAR TIME1TIME2
```

```
48 TC SCALPREP
```

```
49 # STORE SCALER IN MPAC, MPAC+1
```

```
50 TC POSTAND +3
```

```
51 # T1,T2,SCALER NOT COMPATIBLE
```

```
52 EXTEND
```

```
53 # T1,T2 AND SCALER OK
```

```
54 DCS SCALSAVE
```

```
55 DAS MPAC
```

```
56 # FORM DP DIFFERENCE OF POSTSTANDBY SCALER
```

|    |         |        |          |                                            |    |
|----|---------|--------|----------|--------------------------------------------|----|
| 1  |         |        |          |                                            | 1  |
| 2  |         | CAF    | BIT10    | # MINUS PRESTANDBY SCALER AND SHIFT RIGHT  | 2  |
| 3  |         | TC     | SHORTMP  | # 5 TO ALIGN BITS WITH TIME1TIME2.         | 3  |
| 4  |         | CAF    | ZERO     |                                            | 4  |
| 5  |         | TS     | MPAC +2  | # NEEDED FOR TP AGREE                      | 5  |
| 6  |         | TC     | TPAGREE  | # MAKE DP DIFF AGREE                       | 6  |
| 7  |         | CCS    | MPAC     |                                            | 7  |
| 8  |         | TC     | POSTCOM  | # IF DP DIFF NET +, NO SCALER OVERFLOW     | 8  |
| 9  |         | TC     | POSTCOM  | # BETWEEN PRE AND POST STANDBY.            | 9  |
| 10 |         | TC     | +1       | # IF DP DIFF NET -, SCALER OVERFLOWED. ADD | 10 |
| 11 |         | CAF    | BIT10    | # BIT 10 TO HIGH DIFF TO CORRECT.          | 11 |
| 12 |         | ADS    | MPAC     |                                            | 12 |
| 13 | POSTCOM | EXTEND |          | # C(MPAC,+1) IS MAGNITUDE OF DELTA SCALER. | 13 |
| 14 |         | DCA    | TIME2SAV | # PRESTANDBY TIME1TIME2                    | 14 |
| 15 |         | DAS    | MPAC     |                                            | 15 |
| 16 |         | TC     | TPAGREE  | # FORCE SIGN AGREEMENT                     | 16 |
| 17 |         | DXCH   | MPAC     | # UPDATED VALUE FOR T1,T2.                 | 17 |
| 18 |         | DAS    | TIME2    | # LOAD UPDATED VALUE INTO T1,T2, WITH      | 18 |
| 19 |         | TC     | DOWNFLAG | # CLEAR NODOFLAG                           | 19 |
| 20 |         | ADRES  | NODOFLAG |                                            | 20 |
| 21 |         |        |          |                                            | 21 |
| 22 |         | TC     | GOTOP00H |                                            | 22 |
| 23 |         |        |          |                                            | 23 |
| 24 |         |        |          |                                            | 24 |
| 25 |         |        |          |                                            | 25 |
| 26 |         |        |          |                                            | 26 |
| 27 |         |        |          |                                            | 27 |
| 28 |         |        |          |                                            | 28 |
| 29 |         |        |          |                                            | 29 |
| 30 |         |        |          |                                            | 30 |
| 31 |         |        |          |                                            | 31 |
| 32 |         |        |          |                                            | 32 |
| 33 |         |        |          |                                            | 33 |
| 34 |         |        |          |                                            | 34 |
| 35 |         |        |          |                                            | 35 |
| 36 |         |        |          |                                            | 36 |
| 37 |         |        |          |                                            | 37 |
| 38 |         |        |          |                                            | 38 |
| 39 |         |        |          |                                            | 39 |
| 40 |         |        |          |                                            | 40 |
| 41 |         |        |          |                                            | 41 |
| 42 |         |        |          |                                            | 42 |
| 43 |         |        |          |                                            | 43 |
| 44 |         |        |          |                                            | 44 |
| 45 |         |        |          |                                            | 45 |
| 46 |         |        |          |                                            | 46 |
| 47 |         |        |          |                                            | 47 |
| 48 |         |        |          |                                            | 48 |
| 49 |         |        |          |                                            | 49 |
| 50 |         |        |          |                                            | 50 |
| 51 |         |        |          |                                            | 51 |
| 52 |         |        |          |                                            | 52 |
| 53 |         |        |          |                                            | 53 |
| 54 |         |        |          |                                            | 54 |
| 55 |         |        |          |                                            | 55 |
| 56 |         |        |          |                                            | 56 |
| 57 |         |        |          |                                            | 57 |
| 58 |         |        |          |                                            | 58 |
| 59 |         |        |          |                                            | 59 |
| 60 |         |        |          |                                            | 60 |

|    |                     |        |            |    |
|----|---------------------|--------|------------|----|
| 1  | # KEYRUPT1-01 RUPT1 |        |            | 1  |
| 2  |                     | BANK   | 14         | 2  |
| 3  |                     | SETLOC | KEYRUPT    | 3  |
| 4  |                     | BANK   |            | 4  |
| 5  |                     | COUNT* | \$\$/KEYUP | 5  |
| 6  |                     |        |            | 6  |
| 7  | KEYRUPT1            | TS     | BANKRUPT   | 7  |
| 8  |                     | XCH    | Q          | 8  |
| 9  |                     | TS     | QRUPT      | 9  |
| 10 |                     | TC     | LODSAMPT   | 10 |
| 11 |                     | CAF    | LOW5       | 11 |
| 12 |                     | EXTEND |            | 12 |
| 13 |                     | RAND   | MNKEYIN    | 13 |
| 14 | KEYCOM              | TS     | RUPTREG4   | 14 |
| 15 |                     | CS     | FLAGWRD5   | 15 |
| 16 |                     | MASK   | DSKYFBIT   | 16 |
| 17 |                     | ADS    | FLAGWRD5   | 17 |
| 18 |                     |        |            | 18 |
| 19 | ACCEPTUP            | CAF    | CHRPRI0    | 19 |
| 20 |                     | TC     | NOVAC      | 20 |
| 21 |                     | EBANK= | DSPCOUNT   | 21 |
| 22 |                     | 2CADR  | CHARIN     | 22 |
| 23 |                     |        |            | 23 |
| 24 |                     | CA     | RUPTREG4   | 24 |
| 25 |                     | INDEX  | LOCCTR     | 25 |
| 26 |                     | TS     | MPAC       | 26 |
| 27 |                     | TC     | RESUME     | 27 |
| 28 |                     |        |            | 28 |
| 29 |                     |        |            | 29 |
| 30 |                     |        |            | 30 |
| 31 |                     |        |            | 31 |
| 32 |                     |        |            | 32 |
| 33 |                     |        |            | 33 |
| 34 |                     |        |            | 34 |
| 35 |                     |        |            | 35 |
| 36 |                     |        |            | 36 |
| 37 |                     |        |            | 37 |
| 38 |                     |        |            | 38 |
| 39 |                     |        |            | 39 |
| 40 |                     |        |            | 40 |
| 41 |                     |        |            | 41 |
| 42 |                     |        |            | 42 |
| 43 |                     |        |            | 43 |
| 44 |                     |        |            | 44 |
| 45 |                     |        |            | 45 |
| 46 |                     |        |            | 46 |
| 47 |                     |        |            | 47 |
| 48 |                     |        |            | 48 |
| 49 |                     |        |            | 49 |
| 50 |                     |        |            | 50 |
| 51 |                     |        |            | 51 |
| 52 |                     |        |            | 52 |
| 53 |                     |        |            | 53 |
| 54 |                     |        |            | 54 |
| 55 |                     |        |            | 55 |
| 56 |                     |        |            | 56 |
| 57 |                     |        |            | 57 |
| 58 |                     |        |            | 58 |
| 59 |                     |        |            | 59 |
| 60 |                     |        |            | 60 |

## # UPRUPT PROGRAM

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| UPRUPT   | TS     | BANKRUPT |                                            |
|          | XCH    | Q        |                                            |
|          | TS     | QRUPT    |                                            |
|          | TC     | LODSAMPT | # TIME IS SNATCHED IN RUPT FOR NOUN 65.    |
|          | CAF    | ZERO     |                                            |
|          | XCH    | INLINK   |                                            |
|          | TS     | KEYTEMP1 |                                            |
|          | CAF    | BIT3     | # TURN ON UPACT LIGHT                      |
|          | EXTEND |          | # (BIT 3 OF CHANNEL 11)                    |
| UPRPT1   | WOR    | DSALMOUT |                                            |
|          | CAF    | LOW5     | # TEST FOR TRIPLE CHAR REDUNDANCY          |
|          | MASK   | KEYTEMP1 | # LOW5 OF WORD                             |
|          | XCH    | KEYTEMP1 | # LOW5 INTO KEYTEMP1                       |
|          | EXTEND |          |                                            |
|          | MP     | BIT10    | # SHIFT RIGHT 5                            |
|          | TS     | KEYTEMP2 |                                            |
|          | MASK   | LOW5     | # MID 5                                    |
|          | AD     | HI10     |                                            |
|          | TC     | UPTTEST  |                                            |
|          | CAF    | BIT10    |                                            |
|          | EXTEND |          |                                            |
|          | MP     | KEYTEMP2 | # SHIFT RIGHT 5                            |
|          | MASK   | LOW5     | # HIGH 5                                   |
|          | COM    |          |                                            |
|          | TC     | UPTTEST  |                                            |
| UPCK     | CS     | ELRCODE  | # CODE IS GOOD. IF CODE = 'ERROR RESET',   |
|          | AD     | KEYTEMP1 | # CLEAR UPLOCKFL(SET BIT4 OF FLAGWRD7 = 0) |
|          | EXTEND |          | # IF CODE DOES NOT = 'ERROR RESET', ACCEPT |
|          | BZF    | CLUPLOCK | # CODE ONLY IF UPLOCKFL IS CLEAR (=0).     |
|          | CAF    | UPLOCBIT | # TEST UPLOCKFL FOR 0 OR 1                 |
|          | MASK   | FLAGWRD7 |                                            |
|          | CCS    | A        |                                            |
|          | TC     | RESUME   | # UPLOCKFL = 1                             |
|          | TC     | ACCEPTUP | # UPLOCKFL = 0                             |
| CLUPLOCK | CS     | UPLOCBIT | # CLEAR UPLOCKFL (I.E.,SET BIT 4 OF )      |
|          | MASK   | FLAGWRD7 | # FLAGWRD7 = 0)                            |
|          | TS     | FLAGWRD7 |                                            |
|          | TC     | ACCEPTUP |                                            |
| TMFAIL2  | CS     | FLAGWRD7 | # CODE IS BAD                              |
|          | MASK   | UPLOCBIT | # LOCK OUT FURTHER UPLINK ACTIVITY         |
|          | ADS    | FLAGWRD7 | # (BY SETTING UPLOCKFL = 1) UNTIL          |
|          | TC     | RESUME   | # 'ERROR RESET' IS SENT VIA UPLINK.        |
| UPTTEST  | AD     | KEYTEMP1 |                                            |

ELRCODE OCT 22

```
THE RECEPTION OF A BAD CODE(I.E C̄CC FAILURE) LOCKS OUT FURTHER UPLINK ACTIVITY BY SETTING BIT4 OF FLAGWRD7 = 1.
THIS INDICATION WILL BE TRANSFERRED TO THE GROUND BY THE DOWNLINK WHICH DOWNLINKS ALL FLAGWORDS.
WHEN UPLINK ACTIVITY IS LOCKED OUT ,IT CAN BE ALLOWED WHEN THE GROUND UPLINKS AND 'ERROR RESET' CODE.
(IT IS RECOMMENDED THAT THE 'ERROR LIGHT RESET' CODE IS PRECEDED BY 16 BITS THE FIRST OF WHICH IS 1 FOLLOWED
BY 15 ZEROES. THIS WILL ELIMINATE EXTRANEIOUS BITS FROM INLINK WHICH MAY HAVE BEEN LEFT OVER FROM THE ORIGINAL
FAILURE)
UPLINK ACTIVITY IS ALSO ALLOWED(UNLOCKED) DURING FRESH START WHEN FRESH START SETS BIT4 OF FLAGWRD7 = 0.
```

CS XDSPBIT

# DISPLAYS CAN BE CLASSIFIED INTO THE FOLLOWING CATEGORIES --

- # 1. PRIORITY DISPLAYS -- DISPLAYS WHICH TAKE PRIORITY OVER ALL OTHER DISPLAYS. USUALLY THESE DISPLAYS ARE SENT OUT UNDER CRITICAL ALARM CONDITIONS.
- # 2. EXTENDED VERB DISPLAYS -- ALL EXTENDED VERBS AND MARK ROUTINES SHOULD USE EXTENDED VERB (MARK) DISPLAYS.
- # 3. NORMAL DISPLAYS -- ALL MISSION PROGRAM DISPLAYS WHICH INTERFACE WITH THE ASTRONAUT DURING THE NORMAL SEQUENCE OF EVENTS.
- # 4. MISC. DISPLAYS -- ALL DISPLAYS NOT HANDLED BY THE DISPLAY INTERFACE ROUTINES. THESE INCLUDE SUCH DISPLAYS AS MM DISPLAYS AND SPECIAL PURPOSE DISPLAYS HANDLED BY PINBALL.
- # 5. ASTRONAUT INITIATED DISPLAYS -- ALL DISPLAYS INITIATED EXTERNALLY.

# THE FOLLOWING TERMS ARE USED TO DESCRIBE THE STATUS OF DISPLAYS --

- # 1. ACTIVE -- THE DISPLAY WHICH IS (1) BEING DISPLAYED TO THE ASTRONAUT AND WAITING FOR A RESPONSE OR (2) WAITING FIRST IN LINE FOR THE ASTRONAUT TO FINISH USING THE DSKY OR (3) BEING DISPLAYED ON THE DSKY BUT NOT WAITING FOR A RESPONSE.
- # 2. INACTIVE -- A DISPLAY WHICH HAS (1) BEEN ACTIVE BUT WAS INTERRUPTED BY A DISPLAY OF HIGHER PRIORITY, (2) BEEN PUT INTO THE WAITING LIST AT TIME IT WAS REQUESTED DUE TO THE FACT A HIGHER PRIORITY DISPLAY WAS ALREADY DOING, (3) BEEN INTERRUPTED BY THE ASTRONAUT (CALLED A PINBRANCH CONDITION, SINCE THIS TYPE OF INACTIVE DISPLAY IS USUALLY REACTIVATED ONLY BY PINBALL) OR (4) A DISPLAY WHICH HAS FINISHED BUT STILL HAS INFO SAVED FOR RESTART PURPOSES.

# DISPLAY PRIORITIES WORK AS FOLLOWS --

# INTERRUPTS --

- # 1. THE ASTRONAUT CAN INTERRUPT ANY DISPLAY WITH AN EXTERNAL DISPLAY REQUEST.
- # 2. INTERNAL DISPLAYS CAN NOT BE SENT OUT WHEN THE ASTRONAUT IS USING THE DSKY.
- # 3. PRIORITY DISPLAYS INTERRUPT ALL OTHER TYPES OF INTERNAL DISPLAYS. A PRIORITY DISPLAY INTERRUPTING ANOTHER PRIORITY DISPLAY WILL CAUSE AN ABORT UNLESS BIT14 IS SET FOR THE LINUS ROUTINE.
- # 4. A MARK DISPLAY INTERRUPTS ANY NORMAL DISPLAY.
- # 5. A MARK THAT INTERRUPTS A MARK COMPLETELY REPLACES IT.

# ORDER OF WAITING DISPLAYS --

- # 1. ASTRONAUT
- # 2. PRIORITY
- # 3. INTERRUPTED MARK
- # 4. INTERRUPTED NORMAL
- # 5. MARK TO BE REQUESTED (SEE DESCRIPTION OF ENDMARK)
- # 6. MARK WAITING
- # 7. NORMAL WAITING

```
1 # THE DISPLAY ROUTINES ARE INTENDED TO SERVE AS AN INTERFACE BETWEEN THE USER AND PINBALL. THE
2 # FOLLOWING STATEMENTS CAN BE MADE ABOUT NORMAL DISPLAYS AND PRIORITY DISPLAYS (A DESCRIPTION OF MARK ROUTINES
3 # WILL FOLLOW LATER):
4 #
5 # 1. ALL ROUTINES THAT END IN R HAVE AN IMMEDIATE RETURN TO THE USER. FOR ALL FLASHING DISPLAYS THIS RETURN
6 # IS TO THE USER'S CALL CADR +4. FOR THE ONLY NON-FLASHING IMMEDIATE RETURN DISPLAY (GODSPR) THIS RETURN
7 # IS TO THE USER'S CALLING LOC +1.
8 #
9 # 2. ALL ROUTINES NOT ENDING IN R DO NOT DO AN IMMEDIATE RETURN TO THE USER.
10 #
11 # 3. ALL ROUTINES THAT END IN R START A SEPARATE JOB (MAKEPLAY) WITH USER'S JOB PRIORITY.
12 #
13 # 4. ALL ROUTINES NOT ENDING IN R BRANCH DIRECTLY TO MAKEPLAY WHICH MAKES THESE DISPLAYS A PART OF THE
14 # USER'S JOB.
15 #
16 # 5. ALL DISPLAY ROUTINES ARE CALLED VIA BANKCALL.
17 #
18 # 6. TO RESTART A DISPLAY THE USER WILL GENERALLY USE A PHASE OF ONE WITH DESIRED RESTART GROUP (SEE
19 # DESCRIPTION OF RESTARTS).
20 #
21 # 7. ALL FLASHING DISPLAYS HAVE 3 RETURNS TO THE USER FROM ASTRONAUT RESPONSES. A TERMINATE (V34) BRANCHES
22 # TO THE USER'S CALL CADR +1. A PROCEED (V33) BRANCHES TO THE USER'S CALL CADR +2. AN ENTER OR RECYCLE
23 # (V32) BRANCHES TO THE USER'S CALL CADR +3.
24 #
25 # 8. ALL ROUTINES MUST BE USED UNDER EXECUTIVE CONTROL
26
27 # A DESCRIPTION OF EACH ROUTINE WITH AN EXAMPLE FOLLOWS:
28 #
29 # GODSP IS USED TO DISPLAY A VERB NOUN ARRIVING IN A. NO RETURN IS MADE TO THE USER.
30 #
31 # 1. GODSP IS NOT RESTARTABLE
32 # 2. A VERB PASTE WITH GODSP ALWAYS TURNS ON THE FLASH.
33 #
34 # CAF VXXNYY
35 # TC BANKCALL
36 # CADR GODSP
37 # VXXNYY OCT OXXYY
38
39 # GODSPR IS THE SAME AS GODSP ONLY RETURN IS TO THE USER.
40 #
41 # CAF VXXNYY
42 # TC BANKCALL
43 # CADR GODSPR
44 # # IMMEDIATE RETURN OF GODSPR
45
46 # GOFLASH DISPLAYS A FLASHING VERB NOUN WITH NO IMMEDIATE RETURN TO THE USER. 3 RETURNS ARE POSSIBLE FORM
47 # THE ASTRONAUT (SEE NO. 7 ABOVE).
48 #
49 # CAF VXXNYY # VXX NYY WILL BE A FLASHING VERB NOUN.
50 # TC BANKCALL
51 # CADR GOFLASH
52 # # TERMINATE RETURN
53 # # PROCEED RETURN
54 # # ENTER OR RECYCLE RETURN
55
56 # GOPERF1 IS ENTERED WITH DESIRED CHECKLIST VALUE IN A. GOPERF1 WILL DISPLAY THIS VALUE IN R1 BY MEANS OF A
```

```
1 # V01 N25. A FLASHING PLEASE PERFORM ON CHECKLIST (V50 N25) IS THEN DISPLAYED. NO IMMEDIATE RETURN IS MADE TO
2 # USER (SEE NO. 7 ABOVE).
3 #
4 # GOPERF1 BLANKS REGISTERS R2 AND R3
5 # CAF OCTXX # CODE FOR CHECKLIST VALUE XX
6 # TC BANKCALL
7 # CADR GOPERF1
8 # ... # TERMINATE RETURN
9 # ... # PROCEED RETURN
10 # ... # ENTER RETURN
11 # GOPERF2 IS ENTERED WITH A VARIABLE NOUN AND V01 (V00 FOR N10 OR N11) IN A. GOPERF2 WILL FIRST DISPLAY THE
12 # REQUESTED NOUN BY MEANS OF A VO1NYY OR A VOONYY. PLEASE PERFORM ON NOUN (V50 NYY) THEN BECOMES A FLASHING
13 # DISPLAY. NO IMMEDIATE RETURN IS MADE TO THE USER (SEE NO. 7 ABOVE).
14 # GOPERF2 DOES NOT BLANK ANY REGISTERS
15 # CAF VXXNYY # VARIABLE NOUN YY. XX=0 OR 01.
16 # TC BANKCALL
17 # CADR GOPERF2
18 # ... # TERMINATE RETURN
19 # ... # PROCEED RETURN
20 # ... # ENTER RETURN
21 # GOPERF3 IS USED FOR A PLEASE PERFORM ON A PROGRAM NUMBER. THE DESIRED PROGRAM NO. IS ENTERED IN A. GOPERF3
22 # DISPLAYS THE NO. BY MEANS OF A V06 N07 FOLLOWED BY A FLASHING V50 N07 FOR A PLEASE PERFORM. NO IMMEDIATE RETURN
23 # IS MADE TO THE USER (SEE NO. 7 ABOVE).
24 # GOPERF3 BLANKS REGISTERS R2 AND R3
25 # CAF DECXX # REQUEST PERFORM ON PXX
26 # TC BANKCALL
27 # CADR GOPERF3
28 # ... # TERMINATE RETURN
29 # ... # PROCEED RETURN
30 # ... # ENTER RETURN
31 # GOPERF4 IS USED FOR A PLEASE PERFORM ON AN OPTION. THE DESIRED OPTION IS ENTERED IN A AND STORED IN OPTION1.
32 # GOPERF4 DISPLAYS R1 AND R2 BY MEANS OF A V04N06 FOLLOWED BY A FLASHING V50N06 FOR A PLEASE PERFORM. NO
33 # IMMEDIATE RETURN IS MADE TO THE USER (SEE NO. 7 ABOVE).
34 # CAF OCTXX # REQUEST PERFORM ON OPTION XX
35 # TC BANKCALL
36 # CADR GOPERF4
37 # ... # TERMINATE RETURN
38 # ... # PROCEED RETURN
39 # ... # ENTER RETURN
40 # GOPERF4 BLANKS REGISTER R3.
```



# GODSPRET IS USED TO DISPLAY A VERB NOUN ARRIVING IN A WITH A RETURN TO THE USER AFTER THE DISPLAY HAS BEEN SENT OUT.

CAF VXXXNYY  
TC BANKCALL  
CADR GODSPRET

# RETURN TO USER.

# REGODSP IS USED TO DISPLAY A VERB NOUN ARRIVING IN A. REGODSP IS THE SAME AS GODSP ONLY REGODSP REPLACES ANY ACTIVE NORMAL DISPLAY IF ONE WAS ACTIVE.

CAF VXXNYY  
TC BANKCALL  
CADR REGODSP

# REFLASH IS THE SAME AS GOFLASH ONLY REFLASH REPLACES ANY ACTIVE NORMAL DISPLAY IF ONE WAS ACTIVE.

CAF VXXNYY  
TC BANKCALL  
CADR REFLASH

# VXX NYY WILL BE A FLASHING VERB NOUN

# TERMINATE RETURN

# PROCEED RETURN

# ENTER RETURN

# GOFLASHR IF SAME AS GOFLASH ONLY AN IMMEDIATE RETURN IS MADE TO THE USER'S CALL CADR +4.

CAF VXXNYY  
TC BANKCALL  
CADR GOFLASHR

# TERMINATE RETURN

# PROCEED RETURN

# ENTER OR RECYCLE RETURN

# IMMEDIATE RETURN FROM GOFLASHR

# GOPERF1R IS THE SAME AS GOPERF1 ONLY GOPERF1R HAS AN IMMEDIATE RETURN TO USER'S CALL CADR +4.

# GOPERF1R BLANKS REGISTERS R2 AND R3

CAF OCTXX  
TC BANKCALL  
CADR GOPERF1R

# CODE FOR CHECKLIST VALUE XX.

# TERMINATE RETURN

# PROCEED RETURN

# ENTER RETURN

# IMMEDIATE RETURN FROM GOPERF1R

# GOPERF2R IS THE SAME AS GOPERF2 ONLY AN IMMEDIATE RETURN IS MADE TO USER'S CALL CADR +4.

```
1 # GOPERF2R DOES NOT BLANK ANY REGISTERS
2 #
3 # CAF VXXXNYY # VARIABLE NOUN YY REQUESTED. XX=00 OR 01
4 # TC BANKCALL
5 # CADR GOPERF2R
6 # ... # TERMINATE RETURN
7 # ... # PROCEED RETURN
8 # ... # ENTER RETURN
9 # ... # IMMEDIATE RETURN HERE FROM GOPERF2R
10 # GOPERF3R IS THE SAME AS GOPERF3 ONLY AN IMMEDIATE RETURN IS MADE TO USER'S CALL CADR +4.
11 # GOPERF3R BLANKS REGISTERS R2 AND R3
12 # CAF PROGXX # PERFORM PROGRAM XX
13 # TC BANKCALL
14 # CADR GOPERF3R
15 # ... # TERMINATE RETURN
16 # ... # PROCEED RETURN
17 # ... # ENTER RETURN
18 # ... # GOPERF3R IMMEDIATELY RETURNS HERE
19 # GOPERF4R IS THE SAME AS GOPERF4 ONLY AN IMMEDIATE RETURN IS MADE TO USER'S CALL CADR +4.
20 # CAF OCTXX # REQUEST PERFORM ON OPTIONXX
21 # TC BANKCALL
22 # CADR GOPERF4R
23 # ... # TERMINATE RETURN
24 # ... # PROCEED RETURN
25 # ... # ENTER RETURN
26 # ... # IMMEDIATE RETURN TO USER
27 # GOPERF4R BLANKS REGISTER R3.
28 # REFLASHR IS THE SAME AS REFLASH ONLY AN IMMEDIATE RETURN IS MADE TO THE USER'S CALL CADR +4.
29 # CAF VXXNYY # VXX NYY WILL BE A FLASHING VERB NOUN
30 # TC BANKCALL
31 # CADR REFLASHR
32 # ... # TERMINATE RETURN
33 # ... # PROCEED RETURN
34 # ... # ENTER RETURN
35 # ... # IMMEDIATE RETURN TO USER
36 # REGODSPR IS THE SAME AS REGODSP ONLY A RETURN (IMMEDIATE) IS MADE TO THE USER.
```

| # DISPLAY_INTERFACE_ROUTINES |   |      |          | PAGE 1346                  |  |  |    |
|------------------------------|---|------|----------|----------------------------|--|--|----|
| 1                            |   |      |          |                            |  |  | 1  |
| 2                            | # | CAF  | VXXNYY   |                            |  |  | 2  |
| 3                            | # | TC   | BANKCALL |                            |  |  | 3  |
| 4                            | # | CADR | REGODSPR |                            |  |  | 4  |
| 5                            | # | ...  | ...      | # IMMEDIATE RETURN TO USER |  |  | 5  |
| 6                            |   |      |          |                            |  |  | 6  |
| 7                            |   |      |          |                            |  |  | 7  |
| 8                            |   |      |          |                            |  |  | 8  |
| 9                            |   |      |          |                            |  |  | 9  |
| 10                           |   |      |          |                            |  |  | 10 |
| 11                           |   |      |          |                            |  |  | 11 |
| 12                           |   |      |          |                            |  |  | 12 |
| 13                           |   |      |          |                            |  |  | 13 |
| 14                           |   |      |          |                            |  |  | 14 |
| 15                           |   |      |          |                            |  |  | 15 |
| 16                           |   |      |          |                            |  |  | 16 |
| 17                           |   |      |          |                            |  |  | 17 |
| 18                           |   |      |          |                            |  |  | 18 |
| 19                           |   |      |          |                            |  |  | 19 |
| 20                           |   |      |          |                            |  |  | 20 |
| 21                           |   |      |          |                            |  |  | 21 |
| 22                           |   |      |          |                            |  |  | 22 |
| 23                           |   |      |          |                            |  |  | 23 |
| 24                           |   |      |          |                            |  |  | 24 |
| 25                           |   |      |          |                            |  |  | 25 |
| 26                           |   |      |          |                            |  |  | 26 |
| 27                           |   |      |          |                            |  |  | 27 |
| 28                           |   |      |          |                            |  |  | 28 |
| 29                           |   |      |          |                            |  |  | 29 |
| 30                           |   |      |          |                            |  |  | 30 |
| 31                           |   |      |          |                            |  |  | 31 |
| 32                           |   |      |          |                            |  |  | 32 |
| 33                           |   |      |          |                            |  |  | 33 |
| 34                           |   |      |          |                            |  |  | 34 |
| 35                           |   |      |          |                            |  |  | 35 |
| 36                           |   |      |          |                            |  |  | 36 |
| 37                           |   |      |          |                            |  |  | 37 |
| 38                           |   |      |          |                            |  |  | 38 |
| 39                           |   |      |          |                            |  |  | 39 |
| 40                           |   |      |          |                            |  |  | 40 |
| 41                           |   |      |          |                            |  |  | 41 |
| 42                           |   |      |          |                            |  |  | 42 |
| 43                           |   |      |          |                            |  |  | 43 |
| 44                           |   |      |          |                            |  |  | 44 |
| 45                           |   |      |          |                            |  |  | 45 |
| 46                           |   |      |          |                            |  |  | 46 |
| 47                           |   |      |          |                            |  |  | 47 |
| 48                           |   |      |          |                            |  |  | 48 |
| 49                           |   |      |          |                            |  |  | 49 |
| 50                           |   |      |          |                            |  |  | 50 |
| 51                           |   |      |          |                            |  |  | 51 |
| 52                           |   |      |          |                            |  |  | 52 |
| 53                           |   |      |          |                            |  |  | 53 |
| 54                           |   |      |          |                            |  |  | 54 |
| 55                           |   |      |          |                            |  |  | 55 |
| 56                           |   |      |          |                            |  |  | 56 |
| 57                           |   |      |          |                            |  |  | 57 |
| 58                           |   |      |          |                            |  |  | 58 |
| 59                           |   |      |          |                            |  |  | 59 |
| 60                           |   |      |          |                            |  |  | 60 |

```
1 # GOMARK IS USED TO DISPLAY A MARK VERB NOUN ARRIVING IN A. NO RETURN IS MADE TO THE USER.
2 # GOXDSP = GOMARK
3
4 # CAF VXXNYY # VXXNYY CONTAINS VERB AND NOUN
5 # TC BANKCALL
6 # CADR GOMARK # OTHER EXTENDED VERBS USE CADR GOXDSP
7
8 # GOMARKR IS THE SAME AS GOMARK ONLY RETURN IS TO THE USER.
9 # GOXDSPR = GOMARKR
10
11 # CAF VXXNYY
12 # TC BANKCALL
13 # CADR GOMARKR # OTHER EXTENDED VERBS USE CADR GOXDSPR
14 # ... # IMMEDIATE RETURN OF GOMARKR
15
16 # GOMARKF DISPLAYS A FLASHING MARK VERB NOUN WITH NO IMMEDIATE RETURN TO THE USER. 3 RETURNS ARE POSSIBLE FROM
17 # THE ASTRONAUT (SEE NO. 7 ABOVE).
18 # GOXDSPF = GOMARKF
19
20 # CAF VXXNYY # VXXNYY WILL BE A FLASHING MARK VERB NOUN
21 # TC BANKCALL
22 # CADR GOMARKFR # OTHER EXTENDED VERBS USE CADR GOXDSPFR
23
24 # ... # TERMINATE RETURN
25 # ... # PROCEED RETURN
26 # ... # ENTER OR RECYCLE RETURN
27
28 # GOMARKFR IS THE SAME AS GOMARKF ONLY AN IMMEDIATE RETURN IS MADE TO THE USER CALL CADR +4.
29 # GOXDSPFR = GOMARKFR
30
31 # CAF VXXNYY # FLASHING MARK VERB NOUN
32 # TC BANKCALL
33 # CADR GOMARKFR # OTHER EXTENDED VERBS USE CADR GOXDSPFR
34 # ... # TERMINATE RETURN
35 # ... # PROCEED RETURN
36 # ... # ENTER OR RECYCLE RETURN
37 # ... # IMMEDIATE RETURN TO THE USER
38
39 # GOMARK1 IS USED FOR A PLEASE PERFORM ON A MARK REQUEST WITH ONLY 1 ASTRONAUT RETURN TO THE USER. NO IMMEDIATE
40 # RETURN IS MADE. THE DESIRED MARK PLEASE PERFORM VERB AND DESIRED NOUN IS ENTERED IN A. GOMARK1 DISPLAYS R1, R2, R
41 # MEANS OF A V05NYY FOLLOWED BY A FLASHING V5XNYY FOR A PLEASE PERFORM. THE ASTRONAUT WILL RESPOND WITH A MARK
42 # OR MARK REJECT OR AN ENTER. THE ENTER IS THE ONLY ASTRONAUT RESPONSE THAT WILL COME BACK TO THE USER.
43 # CAF V5XNYY # X=1,2,3,4 Y=NOUN
44 # TC BANKCALL
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1 #
2 # CADR GOMARK1
3 # # ENTER RETURN
4 # *** IF BLANKING DESIRED ON NON-R ROUTINES, NOTIFY DISPLAYER.
5 #
6 # GOMARK1R IS THE SAME AS A GOMARK1 ONLY AN IMMEDIATE RETURN IS MADE TO THE USER'S CALL CADR +2.
7 # CAF V5XNYY # X=1,2,3,4 YY=NOUN
8 # TC BANKCALL
9 # CADR GOMARK1R
10 #
11 # # ASTRONAUT ENTER RETURN
12 # # IMMEDIATE RETURN TO USER
13 # GOMARK2 IS THE SAME AS GOMARK1 ONLY 3 RETURNS ARE MADE TO THE USER FROM THE ASTRONAUT.
14 # CAF V5XNYY # X=1,2,3,4 YY=NOUN
15 # TC BANKCALL
16 # CADR GOMARK2
17 #
18 # # TERMINATE RETURN
19 # # PROCEED RETURN
20 # # ENTER RETURN
21 # GOMARK2R IS THE SAME AS GOMARK1R ONLY 3 ASTRONAUT RETURNS ARE MADE TO THE USER.
22 # CAF V5XNYY # X=0,1,2,3,4 YY=NOUN
23 # TCF BANKCALL
24 # CADR GOMARK2R
25 #
26 # # TERMINATE RETURN
27 # # PROCEED RETURN
28 # # ENTER RETURN
29 # GOMARK3 IS USED FOR A PLEASE PERFORM ON A MARK REQUEST WITH A 3 COMP. DEC DISPLAY. THE DESIRED MARK PLEASE
30 # PERFORM VERB AND NOUN ARE ENTERED IN A. GOMARK3 DISPLAYS R1, R2, R3 BY MEANS OF A V06NYY FOLLOWED BY A FLASHING
31 # V5XNYY FOR A PLEASE PERFORM. GOMARK3 HAS 3 ASTRONAUT RETURNS TO THE USER WITH NO IMMEDIATE RETURN.
32 # CAF V5XNYY # X=1,2,3,4 YY=NOUN
33 # TC BANKCALL
34 # CADR GOMARK3
35 #
36 # # TERMINATE RETURN
37 # # PROCEED RETURN
38 # # ENTER RETURN
39 # GOMARK4 IS THE SAME AS GOMARK3 ONLY R2 AND R3 ARE BLANKED AND R1 IS DISPLAYED IN OCTAL.
40 # CAF V5XNYY # X=1,2,3,4 YY=NOUN
41 # TC BANKCALL
42 # CADR GOMARK4
43 #
44 # # TERMINATE RETURN
45 # # PROCEED RETURN
```

```
1 # # ENTER RETURN
2 #
3 # EXDSPRET IS USED TO DISPLAY A VERB NOUN ARRIVING IN A WITH A RETURN MADE TO THE USER AFTER THE DISPLAY HAS BEEN
4 # SEND OUT.
5 # CAF VXNYY
6 # TC BANKCALL
7 # CADR EXDSPRET
8 # # RETURN TO USER
9 # KLEENEX CLEANS OUT ALL MARK DISPLAYS (ACTIVE AND INACTIVE). A RETURN IS MADE TO THE USER AFTER THE MARK DISPLAYS
10 # HAVE BEEN CLEANED OUT.
11 # TC BANKCALL
12 # CADR KLEENEX
13 # # RETURN TO USER
14 # MARKBRAN IS A SPECIAL PURPOSE ROUTINE USED FOR SAVING JOB VAC AREAS (SEE DESCRIPTION OF MARKBRAN BELOW).
15 # TC BANKCALL
16 # CADR MARKBRAN
17 # # BAD RETURN IF MARK DISPLAY NOT ACTIVE
18 # # (GOOD RETURN TO IMMEDIATE RETURN LOC OF
19 # # LAST FLASHING MARK R ROUTINE)
20 # PINBRNCH REESTABLISHES THE LAST ACTIVE FLASHING DISPLAY. IF THERE IS NO ACTIVE FLASHING DISPLAY, THE DSKY IS
21 # BLANKED AND CONTROL IS SENT TO ENDOFJOB.
22 # TC POSTJUMP
23 # CADR PINBRNCH
24 # PRIODSP IS USED AS A PRIORITY DISPLAY. IT WILL DISPLAY A GOFLASH TYPE DISPLAY WITH THREE POSSIBLE RETURNS FROM
25 # THE ASTRONAUT (SEE NO. 7 ABOVE).
26 # THE MAIN PURPOSE OF PRIODSP IS TO REPLACE THE PRESENT DISPLAY WITH A DISPLAY OF HIGHER PRIORITY AND TO
27 # PROVIDE A MEANS FOR RESTORING THE OLD DISPLAY WHEN THE PRIORITY DISPLAY
28 # IS RESPONDED TO BY THE ASTRONAUT.
29 # THE FORMER DISPLAY IS RESTORED BY AN AUTOMATIC BRANCH TO WAKE UP THE DISPLAY THAT WAS INTERRUPTED BY THE
30 # PRIO DISPLAY
31 # CAF VXXNYY # VXXNYY WILL BE A FLASHING VERB NOUN
32 # TC BANKCALL
33 # CADR PRIODSP
34 # # TERMINATE RETURN
35 # # PROCEED RETURN
```

```
1
2 #
3 # PRIODSPR IS THE SAME AS PRIODSP ONLY AN IMMEDIATE RETURN IS MADE TO THE USER'S CALL CADR +4.
4 # CAF VXXNYY # VXXNYY WILL BE A FLASHING VERB NOUN
5 # TC BANKCALL
6 # CADR PRIODSPR
7
8 # # TERMINATE ACTION
9 # # PROCEED RETURN
10 # # ENTER OR RECYCLE RETURN
11 # # IMMEDIATE RETURN
12
13 # PRIOLARM DOES A V05N09 PRIODSPR.
14
15 # CLEANDSP CLEANS OUT ALL NORMAL DISPLAYS (ACTIVE AND INACTIVE). A RETURN IS MADE TO THE USER AFTER NORMAL
16 # DISPLAYS ARE CLEANED OUT.
17 # TC BANKCALL
18 # CADR CLEANDSP
19 # # RETURN TO USER
```

```
1 #
2 #
3 # GENERAL INFORMATION
4 # -----
5 #
6 # ALARM OR ABORT EXIT MODE --
7 # PRIOBORT TC ABORT
8 # OCT 1502
9 #
10 # PRIOBORT IS BRANCHED TO WHEN (1) A NORMAL DISPLAY IS REQUESTED AND ANOTHER NORMAL DISPLAY IS ALREADY ACTIVE
11 # (REFLASH AND REGODSP ARE EXCEPTIONS) OR (2) A PRIORITY DISPLAY IS REQUESTED WHEN ANOTHER PRIORITY DISPLAY IS
12 # ALREADY ACTIVE (A PRIORITY WITH LINUS BIT14 IS AN EXCEPTION).
13 #
14 # ERASABLE INITIALIZATION REQUIRED --
15 # ACCOMPLISHED BY FRESH START -- 1. FLAGWRD4 (USED EXCLUSIVELY BY DISPLAY INTERFACE ROUTINES)
16 # 2. NVSAVE = NORMAL VERB AND NOUN REGISTER.
17 # 3. EBANKTEM = NORMAL INACTIVE FLAGWORD (ALSO CONTAINS NORMALS EBANK).
18 # 5. R1SAVE = MARKBRAN CONTROL WORD
19 # 4. RESTREG = PRIORITY 30 AND SUPERBANK 3.
20 #
21 # OUTPUT --
22 # NVWORD = PRIO VERB AND NOUN
23 # NVWORD +1 (MARKNV) = MARK VERB AND NOUN
24 # NVWORD +2 (NVSAVE) = NORMAL VERB AND NOUN
25 # DSPFLG (EBANKSAV) = PRIO FLAGWORD (INCLUDING EBANK)
26 # DSPFLG +1 (MARKEBAN) = MARK FLAGWORD (INCLUDING EBANK)
27 # DSPFLG +2 (EBANKTEM) = NORMAL FLAGWORD (INCLUDING EBANK)
28 # CADRFLSH = PRIO USER'S CALL CADR +1 LOCATION
29 # CADRFLSH +1 (MARKFLSH) = MARK USER'S CALL CADR +1 LOCATION
30 # CADRFLSH +2 (TEMPFLSH) = NORMAL USER'S CALL CADR +1 LOCATION
31 # PRIOTIME = TIME EACH PRIO REQUEST FIRST SENT OUT
32 # OPTION1 = DESIRED OPTION FROM GOPERF4
33 # FLAGWRD4 = BIT INFO FOR CONTROL OF ALL DISPLAY ROUTINES
34 # DSPTM1 = R1 INFO FOR ASTRONAUT FROM PERFORM DISPLAYS (NORMAL)
35 #
36 # SUBROUTINES USED -- NVSUB, FLAGUP, FLAGDOWN, ENDOFJOB, BLANKSUB, ABORT, JOBWAKE, JOBSLEEP, FINDVAC, PRIOCHNG,
37 # JAMTERM, NVSUBUSY, FLASHON, ENDIDLE, CHANG1, BANKJUMP, MAKECADR, NOVAC
38 #
39 # DEBRIS -- (STORED INTO)
40 # TEMPORARY TEMPORARIES -- A, Q, L, MPAC +2, MPAC +3, MPAC +4, MPAC +5, MPAC +6, RUPREG2, RUPTREG3, CYL,
41 # EBANK, RUPTREG4, LOC, BANKSET, MODE, MPAC, MPAC +1, FACEREG
42 # ERASABLES (SHARED AND USED WITH OTHER PROGRAMS) -- CADRSTOR, DSPLIST, LOC, DSPTM1, OPTION1
43 # ERASABLES (USED ONLY BY DISPLAY ROUTINES) -- NVWORD,+1,+2, DSPFLAG,+1,+2, CADRFLSH,+1,+2, PRIOTIME, FLAGWRD4,
```



# R1SAVE, MARK2PAC

# DEBRIS -- (USED BUT NOT STORED INTO) -- NOUNREG, VERBREG, LOCCTR, MONSAVE1

# FLAGWORD DESCRIPTIONS --

# FLAGWRD4 -- SEE DESCRIPTION UNDER LOG SECTION ERASABLE ASSIGNMENTS

# DSPFLG, DSPFLG+1, DSPFLG+2

# -----

# BITS 1 BLANK R1

# 2 BLANK R2

# 3 BLANK R3

# 4 FLASHING DISPLAY REQUESTED

# 5 PERFORM DISPLAY REQUESTED

# 6 ----- EXDSPRET GODSPRET

# 7 PRIO DISPLAY -----

# 8 ----- DEC MARK PERFORM -----

# 9 EBANK

# 10 EBANK

# 11 EBANK

# 12 ----- V99PASTE

# 13 2ND PART OF PERFORM

# 14 REFLASH OR REDO ----- REFLASH OR REDO

# 15 ----- MARK REQUEST -----

#

# RESTARTING DISPLAYS --

#

# RULES FOR THE DSKY OPERATOR --

# 1. PROCEED AND TERMINATE SERVE AS RESPONSES TO REQUESTS FOR OPERATOR RESPONSE (FLASHING V/N). AS LONG

# AS THERE IS ANY REQUEST AWAITING OPERATOR RESPONSE, ANY USE OF PROCEED OR TERMINATE WILL SERVE AS  
# RESPONSES TO THAT REQUEST. CARE SHOULD BE EXERCISED IN ATTEMPTING TO KILL AN OPERATOR INITIATED MONITOR  
# WITH PROCEED AND TERMINATE FOR THIS REASON.# 2. THE ASTRONAUT MUST RESPOND TO A PRIORITY DISPLAY NO SOONER THAN 2 SECONDS FROM THE TIME THE  
# PROGRAM SENT OUT THE REQUEST FOR OPERATOR RESPONSE (THE ASTRONAUT WOULD SEE THIS DISPLAY FOR LESS TIME  
# DUE TO TIME IT TAKES TO GET DISPLAY SENT OUT.) IF THE ASTRONAUT RESPONDS TOO SOON, THE PRIORITY DISPLAY  
# IS SENT OUT AGAIN -- AND AGAIN UNTIL AN ACCUMULATED 2 SECS FROM THE TIME THE FIRST PRIORITY DISPLAY  
# OUT. THE SAME 2 SEC. DELAY WILL OCCUR AT 163.84 SECS OR IN ANY MULTIPLE OF THAT TIME DUE TO PROGRAM  
# CONSIDERATION.

# 3. KEY RELEASE BUTTON --

# A) IF THE KEY RELEASE LIGHT IS ON, IT SIMPLY RELEASES THE KEYBOARD AND DISPLAY FOR INTERNAL USE.

# B) IF THE KEY RELEASE LIGHT IS OFF, AND IF SOME REQUEST FOR OPERATOR RESPONSE (FLASHING V/N) IS STILL  
# AWAITING RESPONSE THEN IT RE-ESTABLISHES THE DISPLAYS THAT ORIGINALLY REQUESTED RESPONSE.# IF AN OPERATOR WANTS THEREFORE TO RE-ESTABLISH BUT CONDITION (A) IS ENCOUNTERED, A SECOND DEPRESSION OF  
# KEY RELEASE BUTTON MAY BE NECESSARY.

# 4. IT IS IMPORTANT TO ANSWER ALL REQUESTS FOR OPERATOR RESPONSE.

# 5. IT IS ALWAYS GOOD PRACTICE TO TERMINATE AN EXTENDED VERB BEFORE ASKING FOR ANOTHER ONE OR THE SAME ONE  
# OVER AGAIN.

#

# SPECIAL CONSIDERATONS --

1412 THE

```
1 # CALLING SEQUENCE FOR BLANKING
2 #
3 # CAF BITX # X=1,2,3 BLANK R1,R2,R3 RESPECTIVELY
4 # TC BLANKET
5 # # RETURN TO USER HERE
6 # IN ORDER TO USE BLANKET CORRECTLY, THE USER MUST USE A DISPLAY ROUTINE THAT ENDS IN R FIRST FOLLOWED BY THE CALL
7 # TO BLANKET AT THE IMMEDIATE RETURN LOC.
8 BLOCK 02
9 SETLOC FFTAG4
10 BANK
11
12 COUNT* $$/DSPLA
13 BLANKET TS MPAC +6
14 CS PLAYTEM4
15 MASK MPAC +6
16 INDEX MPAC +5
17 ADS PLAYTEM4
18
19 TC Q
20
21 ENDMARK TC POSTJUMP
22 CADR MARKEND
23
24 CLEARMRK CAF ZERO
25 TS EXTVBACT
26
27 +2 INHINT
28 CS XDSPBIT
29 MASK FLAGWRD4
30 TS FLAGWRD4
31
32 RELINT
33 TC Q
34
35 # *** ALL EXTENDED VERB ROUTINES THAT HAVE AT LEAST ONE FLASHING DISPLAY MUST TCF ENDMARK OR TCF ENDEXT WHEN
36 # FINISHED.
37
38 BANK 10
39 SETLOC DISPLAYS
40 BANK
41
42 COUNT* $$/DSPLA
43
44 # NTERONLY IS USED TO DIFFERENTIATE THE MARK ROUTINE WITH ONLY ONE RETURN TO THE USER FROM THE MARKING ROUTINE WITH
45 # 3 RETURNS TO THE USER. THIS ROUTINE IS ONLY USED BY GOMARK1 AND GOMARK1R.
46
47 MARKEND TC CLEARMRK
48 TCF MARKOVER
49
50 GOMARK TS PLAYTEM1 # ENTRANCE FOR MARK GODSP
51
52
53
54
55
56
57
58
59
60
```

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  | GOMARS   | CAF    | BIT15    | # BIT15 SET FOR ALL MARK REQUESTS          | 2  |
| 3  |          | TCF    | GOFLASH2 |                                            | 3  |
| 4  |          |        |          |                                            | 4  |
| 5  | KLEENEX  | CAF    | ZERO     | # CLEAN OUT EXTENDED VERBS                 | 5  |
| 6  | GOMARKF  | TS     | PLAYTEM1 | # ENTRANCE FOR MARK GOFLASH                | 6  |
| 7  |          |        |          |                                            | 7  |
| 8  |          | CAF    | MARKFMSK | # MARK, FLASH                              | 8  |
| 9  |          | TCF    | GOFLASH2 |                                            | 9  |
| 10 |          |        |          |                                            | 10 |
| 11 | GOMARK2  | TS     | PLAYTEM1 | # MARK GOPERFS-3 AST. RETURNS              | 11 |
| 12 | MARKFORM | CAF    | MPERFMSK | # MARK, PERFORM, FLASH                     | 12 |
| 13 |          | TCF    | GOFLASH2 |                                            | 13 |
| 14 |          |        |          |                                            | 14 |
| 15 | GOMARK3  | TS     | PLAYTEM1 | # USED FOR 3COMP DECIMAL PERFORM           | 15 |
| 16 |          | CAF    | MARK3MSK |                                            | 16 |
| 17 |          | TCF    | GOFLASH2 |                                            | 17 |
| 18 |          |        |          |                                            | 18 |
| 19 | GOMARK4  | TS     | PLAYTEM1 |                                            | 19 |
| 20 |          | CAF    | MARK4MSK | # MARK,PERFORM,FLASH,BLANK                 | 20 |
| 21 |          | TCF    | GOFLASH2 |                                            | 21 |
| 22 |          |        |          |                                            | 22 |
| 23 | GOMARKR  | TS     | PLAYTEM1 | # ENTRANCE FOR MARK GODSPR                 | 23 |
| 24 |          |        |          |                                            | 24 |
| 25 |          | CAF    | BIT15    |                                            | 25 |
| 26 |          | TCF    | GODSPR2  |                                            | 26 |
| 27 |          |        |          |                                            | 27 |
| 28 | GOMARKFR | TS     | PLAYTEM1 | # ENTRANCE FOR MARK GOFLASHR               | 28 |
| 29 |          |        |          |                                            | 29 |
| 30 |          | CAF    | MARKFMSK |                                            | 30 |
| 31 |          | TCF    | GODSPRS  |                                            | 31 |
| 32 |          |        |          |                                            | 32 |
| 33 | GOMARK2R | TS     | PLAYTEM1 | # MARK GOPERFS-3 AST. RETS+ IMMEDIATE RET. | 33 |
| 34 |          | CAF    | MPERFMSK | # MARK, PERFORM, FLASH                     | 34 |
| 35 |          | TCF    | GODSPRS  |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 | GOMARK3R | TS     | PLAYTEM1 |                                            | 37 |
| 38 |          | CAF    | MARK3MSK |                                            | 38 |
| 39 |          | TCF    | GODSPRS  |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 | MAKEMARK | CAF    | ONE      |                                            | 41 |
| 42 |          | TC     | COPIES   |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          | CA     | FLAGWRD4 | # IS NORM OR PRIO BUSY OR WAITING          | 44 |
| 45 |          | MASK   | OCT34300 |                                            | 45 |
| 46 |          | CCS    | A        |                                            | 46 |
| 47 |          | TCF    | CHKPRIO  |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          | CA     | FLAGWRD4 | # IS MARK SLEEPING DUE TO ASTRO BUSY       | 49 |
| 50 |          | MASK   | MRKNVBIT |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          | EXTEND |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

```
1 BZF MARKPLAY # NO
2
3
4 TCF ENDOFJOB
5
6 MARKPLAY INHINT
7
8 CS FIVE # RESET MARK OVER NORM, SET MARK
9 MASK FLAGWRD4
10 AD ONE
11
12 TS FLAGWRD4
13 RELINT
14
15 GOGOMARK CS MARKFLAG # PERFORM
16 MASK BIT5
17 CCS A
18
19 TCF MARKCOP
20 CS MARKNV
21 TS MARKNV
22
23 MARKCOP CAF ONE # MARK INDEX
24 TCF PRIOPLAY
25
26 COPYTOGO CA MPAC2SAV
27 TS MPAC +2
28
29 COPYPACS INDEX COPINDEX
30 CAF PRIOOCT
31 TS GENMASK
32
33 INDEX COPINDEX
34 CAF EBANKSAV
35 TS TEMPOR2 # ACTIVE EBANK AND FLAG
36
37 TS EBANK
38
39 TC Q
```

```
40 # PINCHEK CHECKS TO SEE IF THE CURRENT MARK REQUEST IS MADE BY THE ASTRONAUT WHILE INTERRUPTING A GOPLAY DISPLAY
41 # (A NORMAL OR A PRIO). IF THE ASTRONAUT TRIES TO MARK DURING A PRIO, THE CHECK FAIL LIGHT GOES ON AND THE MARK
42 # REQUEST IS ENDED. IF HE TRIES TO MARK DURING A NORM, THE MARK IS ALLOWED. IN THIS CASE THE NORM IS PUT TO SLEEP
43 # UNTIL ALL MARKING IS FINISHED.
44 #
45 # IF THE MARK REQUEST COMES FROM THE PROGRAM DURING A TIME THE ASTRONAUT IS NOT INTERRUPTING A NORMAL OR A
46 # PRIO, THE MARK REQUEST IS PUT TO SLEEP UNTIL THE PRESENT ACTIVE DISPLAY IS RESPONDED TO BY THE ASTRONAUT.
```

```
47 CHKPRIO CA FLAGWRD4 # MARK ATTEMPT DURING PRIO
48 MASK OCT24100
49 CCS A
50
51 TCF MARSLEEP
52
53 CS FLAGWRD4
```

```
1
2 MASK MKOVBIT # SET MARK OVER NORM
3 INHINT
4 ADS FLAGWRD4
5
6 TCF SETNORM
7
8 MARKPERF CA MARKNV
9 MASK VERBMASK
10 TCF NV50DSP
11
12 GODSP TS PLAYTEM1
13
14 GODSP2 CAF ZERO
15 TCF GOFLASH2
16
17 GODSPRET TS PLAYTEM1 # ENTRANCE FOR A GODSP WITH A PASTE
18
19 CAF BIT6 # SET BIT6 TO GO BACK TO USER AFTER NVSUB
20 TCF GOFLASH2
21
22 GODSPR TS PLAYTEM1
23
24 GODSPR1 CAF ZERO
25 GODSPR2 TS PLAYTEM4
26
27 CAF ZERO # * DON'T MOVE
28 TCF GODSPRS1
29
30 # CLEANDSP IS USED FOR CLEARING OUT A NORMAL DISPLAY THAT IS PRESENTLY ACTIVE OR A NORMAL DISPLAY THAT IS
31 # SET UP TO BE STARTED OR RESTARTED.
32 #
33 # NORMALLY THE USER WILL NOT NEED TO USE THIS ROUTINE SINCE A NEW NORMAL DISPLAY AUTOMATICALLY CLEARS OUT AN
34 # OLD DISPLAY.
35 #
36 # CALLING SEQUENCE FOR CLEANDSP --
37 #
38 # TC BANKCALL
39 # CADR CLEANDSP
40
41 CLEANDSP CAF ZERO
42 REFLASH TS PLAYTEM1
43
44 CAF REDOMASK # FLASH AND PERMIT
45 TCF GOFLASH2
46
47 REFLASHR TS PLAYTEM1
48
49 CAF REDOMASK # FLASH AND PERMIT
50 TCF GODSPRS
51
52
53
54
55
56
57
```

REGODSP TS PLAYTEM1

CAF BIT14  
TCF GOFLASH2

REGODSPR TS PLAYTEM1

CAF BIT14  
TCF GODSPR2

CLOCPLAY TS PLAYTEM1

CAF CLOCKCON  
TCF GOFLASH2

GOFLASH TS PLAYTEM1

CAF BIT4 # LEAVE ONLY FLASH BIT SET

GOFLASH2 TS PLAYTEM4

TC SAVELOCS

RELINT

TCF MAKEPLAY # BRANCH DIRECT WITH NO SEPARATE JOB CALL

PRIODSPR TS PLAYTEM1

CAF BITS7+4  
TCF GODSPRS

PRIODSP TS PLAYTEM1

SETPRIO CAF BITS7+4

TCF GOFLASH2

MAKEPRIO CAF ZERO

TS COPINDEX

TC LINUSCHR  
TCF HIPRIO # LINUS RETURN  
CA FLAGWRD4

MASK OCT20100 # IS PRIO IN ENDIDLE OR BUSY

CCS A  
TCF PRIOBORT # YES, ABORT

HIPRIO CA FLAGWRD4 # MARK ACTIVE

MASK OCT40400

EXTEND  
BZF ASKIFNRM # NO

|          |               |                      |                                           |
|----------|---------------|----------------------|-------------------------------------------|
| SETMARK  | CAF<br>TCF    | ZERO<br>JOBXCHS      |                                           |
| ASKIFNRM | CA<br>MASK    | FLAGWRD4<br>OCT10200 | # NORMAL ACTIVE<br># BITS 13+8            |
|          | EXTEND<br>BZF | OKTOCOPY             | # NO                                      |
| SETNORM  | CAF<br>TCF    | ONE<br>JOBXCHS       |                                           |
| OKTOCOPY | TC<br>TC      | COPYNORM<br>WITCHONE |                                           |
|          | TC            | JOBWAKE              |                                           |
|          | TC            | XCHTOEND             |                                           |
| REDOPRIO | CA<br>TS      | TIME1<br>PRIOTIME    | # SAVE TIME PRIODSP SENT OUT              |
| KEEPPRIO | CAF<br>TCF    | ZERO<br>PRIOPLAY     | # START UP PRIO DISPLAY                   |
| MAKEPLAY | CA<br>MASK    | PRIORITY<br>PRIO37   | # SAVE USER'S PRIORITY                    |
|          | TS            | USERPRIO             |                                           |
|          | CAF<br>TC     | PRIO33<br>PRIOCHNG   | # RAISE PRIORITY FOR FAST JOBS AFTER WAKE |
|          | CA<br>MASK    | PLAYTEM4<br>BITS15+7 | # IS IT MARK OR PRIO OR NORM              |
|          | CCS<br>TCF    | A<br>MAKEPRIO        | # ITS PRIO                                |
|          | TCF<br>TCF    | IFLEGAL<br>MAKEMARK  | # ITS MARK                                |
| IFLEGAL  | CAF<br>TS     | TWO<br>COPINDEX      |                                           |
|          | TC            | LINUSCHR             |                                           |
|          | TCF           | OKTOPLAY             | # LINUS RETURN                            |
|          | CS<br>MASK    | EBANKTEM<br>BIT4     |                                           |
|          | CCS<br>TCF    | A<br>OKTOPLAY        | # NO                                      |
|          | CA            | FLAGWRD4             | # WAS NORM ASLEEP                         |



```
1
2 MASK NBUSMASK # ARE ANY NORMS ASLEEP
3 EXTEND
4 BZF OKTOPLAY # NO
5
6 PRIOBORT TC POOD00
7 OCT 1502
8
9 OKTOPLAY TC COPIES2
10
11 CA USERPRIO
12 EXTEND
13 ROR SUPERBNK
14 TS RESTREG
15
16 CA FLAGWRD4 # PRIO OR MARK GOING
17 MASK PMMASK
18 CCS A
19 TCF GOSLEEPS # MARK GOING
20
21 TCF +2
22 TCF GOSLEEPS
23
24 # COULD PUT NORM BUSY CHECK HERE TO SAVE TIME
25
26 TC WITCHONE # IS IT NVSUB BUSY, ENDIDLE OR NOONE
27 TC JOBWAKE
28
29 TC XCHTOEND
30
31 PLAYJUM1 CAF TWO
32 PRIOPLAY TS COPINDEX
33
34 TCF GOPLAY
35
36 EXDSPRET TS PLAYTEM1
37
38 CAF BIT15+6
39 TCF GOFLASH2
40
41 GOPERF1 TS NORMTEM1 # STORE DESIRED CHECKLIST VALUE
42 CAF VO1N25 # USED TO DISPLAY CHECKLIST VALUE IN R1
43
44 GOPERFS TS PLAYTEM1
45
46 CAF PERFMASK # LEAVE ONLY FLASH, PERFORM, BLANKING
47 TCF GOFLASH2
48
49 GOPERF2 TS PLAYTEM1 # DESIRED VERB-NOUN TO DISPLAY R1,R2,R3
50
51 CAF PERF2MSK
52 TCF GOFLASH2
53
54
55
56
57
58
59
60
```

|          |                                        |                                            |                                                               |
|----------|----------------------------------------|--------------------------------------------|---------------------------------------------------------------|
| GOPERF4  | TC                                     | PURRS4                                     |                                                               |
|          | TCF                                    | GOFLASH2                                   |                                                               |
| GOFLASHR | TS                                     | PLAYTEM1                                   |                                                               |
| GODSPRS  | CAF<br>TS                              | BIT4<br>PLAYTEM4                           | # LEAVE ONLY FLASH BIT SET                                    |
|          | CAF                                    | THREE                                      |                                                               |
| GODSPRS1 | INHINT<br>TS                           | RUPTREG3                                   | # IMMEDIATE RETURN IS CALL CADR +4                            |
|          | CA<br>MASK<br>TS                       | PRIORITY<br>PRIO37<br>NEWPRIO              | # MAKE DISPLAY ONE HIGHER THAN USER                           |
|          | CA<br>MASK                             | PLAYTEM4<br>BIT4                           | # IS THIS A FLASHING R DISPLAY                                |
|          | CCS<br>TCF<br>CA                       | A<br>VACDSP<br>NEWPRIO                     | # YES, MAKE DSPLAY JOB A VAC<br># NO, MAKE DSPLAY JOB A NOVAC |
|          | TC<br>EBANK=<br>2CADR                  | NOVAC<br>WHOCARES<br>MAKEPLAY              |                                                               |
|          | TCF                                    | BOTHJOBS                                   |                                                               |
| VACDSP   | CA<br>EXTEND<br>ROR<br>TS<br>CAF<br>TC | BBANK<br>SUPERBNK<br>L<br>MAKEGEN<br>SPVAC |                                                               |
| BOTHJOBS | TC                                     | SAVELOCS                                   | # COPY TEMPS INTO PERMANENT REGISTERS                         |
|          | EXTEND<br>DCA<br>INDEX                 | MPAC<br>LOCCTR                             | +1<br># SAVE NVWORD AND USER'S MPAC +2                        |
|          | DXCH                                   | MPAC                                       | +1                                                            |
|          | EXTEND                                 |                                            | # SAVE USER'S CADR, FLAGS AND EBANK                           |
|          | DCA<br>INDEX<br>DXCH                   | MPAC<br>LOCCTR<br>MPAC                     | +3<br>+3                                                      |
|          | CA<br>TS                               | LOCCTR<br>MPAC                             | +5                                                            |

|    |          |        |          |                                         |
|----|----------|--------|----------|-----------------------------------------|
| 1  |          |        |          |                                         |
| 2  |          | TC     | SAVELOCR |                                         |
| 3  |          | RELINT |          |                                         |
| 4  |          |        |          |                                         |
| 5  |          | TCF    | BANKJUMP | # CALL CADR +4                          |
| 6  |          |        |          |                                         |
| 7  | GOPERF1R | TS     | NORMTEM1 | # DESIRED CHECKLIST VALUE               |
| 8  |          |        |          |                                         |
| 9  |          | CAF    | V01N25   | # DISPLAYS CHECKLIST VALUE IN R1        |
| 10 |          |        |          |                                         |
| 11 | GOPERFRS | TS     | PLAYTEM1 |                                         |
| 12 |          |        |          |                                         |
| 13 |          | CAF    | PERFMASK | # LEAVE ONLY FLASH, PERFORM, BLANKING   |
| 14 |          | TCF    | GODSPRS  |                                         |
| 15 |          |        |          |                                         |
| 16 | GOPERF2R | TS     | PLAYTEM1 | # DESIRED VERB-NOUN TO DISPLAY R1,R2,R3 |
| 17 |          |        |          |                                         |
| 18 |          | CAF    | PERF2MSK |                                         |
| 19 |          | TCF    | GODSPRS  |                                         |
| 20 |          |        |          |                                         |
| 21 | GOPERF4R | TC     | PURRS4   |                                         |
| 22 |          |        |          |                                         |
| 23 |          | TCF    | GODSPRS  |                                         |
| 24 |          |        |          |                                         |
| 25 | PURRS4   | TS     | OPTION1  | # DESIRED OPTION CODE                   |
| 26 |          |        |          |                                         |
| 27 |          | CAF    | V04N06   |                                         |
| 28 |          | TS     | PLAYTEM1 |                                         |
| 29 |          |        |          |                                         |
| 30 |          | CAF    | PERF4MSK | # FLASH, PERFORM AND EBANK R3           |
| 31 |          | TC     | Q        |                                         |
| 32 |          |        |          |                                         |
| 33 | SAVELOCS | INHINT |          |                                         |
| 34 |          |        |          |                                         |
| 35 |          | CS     | OCT3400  | # EBANK BITS                            |
| 36 |          | MASK   | PLAYTEM4 |                                         |
| 37 |          | AD     | EBANK    |                                         |
| 38 |          | TS     | PLAYTEM4 |                                         |
| 39 |          |        |          |                                         |
| 40 | SAVELOCR | LXCH   | Q        |                                         |
| 41 |          |        |          |                                         |
| 42 |          | TC     | MAKECADR |                                         |
| 43 |          | TS     | PLAYTEM3 |                                         |
| 44 |          |        |          |                                         |
| 45 |          | AD     | RUPTREG3 | # NOT USED FOR NON R ROUTINES           |
| 46 |          | TC     | L        |                                         |
| 47 |          |        |          |                                         |
| 48 | COPYNORM | CAF    | ZERO     |                                         |
| 49 | COPIES   | TS     | COPINDEX |                                         |
| 50 | COPIES2  | INHINT |          |                                         |
| 51 |          | CA     | PLAYTEM4 | # FLAGWORD                              |
| 52 |          |        |          |                                         |
| 53 |          |        |          |                                         |
| 54 |          |        |          |                                         |
| 55 |          |        |          |                                         |
| 56 |          |        |          |                                         |
| 57 |          |        |          |                                         |
| 58 |          |        |          |                                         |
| 59 |          |        |          |                                         |
| 60 |          |        |          |                                         |

|    |          |        |                                         |    |
|----|----------|--------|-----------------------------------------|----|
| 1  |          |        |                                         | 1  |
| 2  |          | INDEX  | COPINDEX                                | 2  |
| 3  |          | TS     | EBANKSAV                                | 3  |
| 4  |          |        | # EQUIV TO DSPFLG                       | 4  |
| 5  |          | MASK   | CADRMASK                                | 5  |
| 6  |          | EXTEND | # FLASH AND GODSPRET                    | 6  |
| 7  |          | BZF    | SKIPADD                                 | 7  |
| 8  |          |        |                                         | 8  |
| 9  |          | CA     | PLAYTEM3                                | 9  |
| 10 |          | INDEX  | COPINDEX                                | 10 |
| 11 |          | TS     | CADRFLSH                                | 11 |
| 12 |          |        |                                         | 12 |
| 13 | SKIPADD  | CA     | PLAYTEM1                                | 13 |
| 14 |          | INDEX  | COPINDEX                                | 14 |
| 15 |          | TS     | NVWORD                                  | 15 |
| 16 |          |        | # VERB NOUN                             | 16 |
| 17 |          | TCF    | RELINTQ                                 | 17 |
| 18 |          |        |                                         | 18 |
| 19 | GOSLEEPS | INDEX  | COPINDEX                                | 19 |
| 20 |          | CA     | PRIOOCT                                 | 20 |
| 21 |          | MASK   | WAITMASK                                | 21 |
| 22 |          | TC     | UPENT2                                  | 22 |
| 23 | WAITMASK | OCT    | 3004                                    | 23 |
| 24 |          | CS     | ONE                                     | 24 |
| 25 |          | AD     | COPINDEX                                | 25 |
| 26 |          | TS     | FACEREG                                 | 26 |
| 27 |          |        |                                         | 27 |
| 28 | XCHSLEEP | INDEX  | FACEREG                                 | 28 |
| 29 |          | CAF    | WAKECADR                                | 29 |
| 30 |          | INHINT |                                         | 30 |
| 31 |          | TC     | JOBWAKE                                 | 31 |
| 32 |          |        | # FIND CADR IN JOB AREA                 | 32 |
| 33 |          | TC     | XCHTOEND                                | 33 |
| 34 |          |        | # CAUSES AWAKENED JOB TO GO TO ENDOFJOB | 34 |
| 35 |          | INDEX  | FACEREG                                 | 35 |
| 36 |          | CAF    | WAKECADR                                | 36 |
| 37 |          | TCF    | JOBSLEEP                                | 37 |
| 38 |          |        | # REPLACE SAME CADR BUT NEW JOB AREA    | 38 |
| 39 | JOBXCHS  | TS     | FACEREG                                 | 39 |
| 40 |          |        | # CONTROLS TYPE OF DISPLAY PUT TO SLEEP | 40 |
| 41 |          | TC     | WITCHONE                                | 41 |
| 42 |          | TC     | JOBWAKE                                 | 42 |
| 43 |          | CA     | FACEREG                                 | 43 |
| 44 |          | INDEX  | LOCCTR                                  | 44 |
| 45 |          | TS     | FACEREG                                 | 45 |
| 46 |          |        |                                         | 46 |
| 47 |          | CAF    | XCHQADD                                 | 47 |
| 48 |          | TC     | XCHNYLOC                                | 48 |
| 49 |          |        |                                         | 49 |
| 50 |          | INDEX  | FACEREG                                 | 50 |
| 51 |          | CA     | MARKOCT                                 | 51 |
| 52 |          | MASK   | IDLESLEP                                | 52 |
| 53 |          |        |                                         | 53 |
| 54 |          |        |                                         | 54 |
| 55 |          |        |                                         | 55 |
| 56 |          |        |                                         | 56 |
| 57 |          |        |                                         | 57 |
| 58 |          |        |                                         | 58 |
| 59 |          |        |                                         | 59 |
| 60 |          |        |                                         | 60 |

|    |                                                                                                                |        |          |                                           |
|----|----------------------------------------------------------------------------------------------------------------|--------|----------|-------------------------------------------|
| 1  |                                                                                                                |        |          |                                           |
| 2  |                                                                                                                | TC     | DOWNENT2 |                                           |
| 3  | IDLEMASK                                                                                                       | OCT    | 74004    | # * DON'T MOVE                            |
| 4  |                                                                                                                |        |          |                                           |
| 5  |                                                                                                                | INDEX  | FACEREG  | # BIT SHOWS PRIO INTERRUPTED NORM OR MARK |
| 6  |                                                                                                                | CA     | BIT5     | # BIT5 FOR MARK, BIT4 FOR NORMAL          |
| 7  |                                                                                                                | AD     | FOUR     |                                           |
| 8  |                                                                                                                | TC     | UPENT2   | # FLAG ROUTINE DOES RELINT                |
| 9  | XCHQADD                                                                                                        | GENADR | XCHSLEEP | # * DON'T MOVE                            |
| 10 |                                                                                                                | CA     | FLAGWRD4 |                                           |
| 11 |                                                                                                                | MASK   | MKOVBIT  | # MARK OVER NORM?                         |
| 12 |                                                                                                                | CCS    | A        |                                           |
| 13 | GENMARK                                                                                                        | TC     | MARKPLAY | # USED AS GENADR FOR JOBWAKE              |
| 14 |                                                                                                                | TCF    | OKTOCOPY |                                           |
| 15 |                                                                                                                |        |          |                                           |
| 16 | MARKWAKE                                                                                                       | CAF    | ZERO     |                                           |
| 17 | WAKEPLAY                                                                                                       | TS     | TEMPOR2  |                                           |
| 18 |                                                                                                                |        |          |                                           |
| 19 |                                                                                                                | INDEX  | TEMPOR2  |                                           |
| 20 |                                                                                                                | CA     | BITS5+11 |                                           |
| 21 |                                                                                                                | AD     | FOUR     |                                           |
| 22 |                                                                                                                | TC     | DOWNENT2 |                                           |
| 23 | MARKFMSK                                                                                                       | OCT    | 40010    | # *** DON'T MOVE                          |
| 24 |                                                                                                                |        |          |                                           |
| 25 |                                                                                                                | INDEX  | TEMPOR2  |                                           |
| 26 |                                                                                                                | CAF    | WAKECADR |                                           |
| 27 |                                                                                                                | INHINT |          |                                           |
| 28 |                                                                                                                | TC     | JOBWAKE  |                                           |
| 29 |                                                                                                                |        |          |                                           |
| 30 |                                                                                                                | TCF    | ENDRET   |                                           |
| 31 |                                                                                                                |        |          |                                           |
| 32 | # ALL .1 RESTARTS BRANCH DIRECTLY TO INITDSP. NORMAL DISPLAYS ARE THE ONLY DISPLAYS ALLOWED TO USE .1 RESTARTS |        |          |                                           |
| 33 | # INITDSP FIRST RESTORES THE EBANK AND THE SUPERBANK TO THE MOST RECENT NORMAL EBANK AND SUPERBANK.            |        |          |                                           |
| 34 | #                                                                                                              |        |          |                                           |
| 35 | # IF THE MOST RECENT NORMAL DISPLAY REQUEST WAS NOT FINISHED, CONTROL IS SENT BACK TO THE LAST NORMAL USER.    |        |          |                                           |
| 36 | # OTHERWISE THE NORMAL DISPLAY SET UP IN THE NORMAL DISPLAY REGS IS STARTED UP IMMEDIATELY.                    |        |          |                                           |
| 37 |                                                                                                                |        |          |                                           |
| 38 | INITDSP                                                                                                        | CA     | EBANKTEM | # RESTORE MOST RECENT NORMAL EBANK        |
| 39 |                                                                                                                | TS     | EBANK    |                                           |
| 40 |                                                                                                                |        |          |                                           |
| 41 |                                                                                                                | CA     | RESTREG  | # SUPERBANK AND JOB PRIORITY              |
| 42 |                                                                                                                | TC     | SUPERSW  | # RESTORE SUPERBANK                       |
| 43 |                                                                                                                |        |          |                                           |
| 44 |                                                                                                                | MASK   | PRI037   |                                           |
| 45 |                                                                                                                | TC     | PRI0CHNG |                                           |
| 46 |                                                                                                                |        |          |                                           |
| 47 |                                                                                                                | CS     | THREE    |                                           |
| 48 |                                                                                                                | AD     | TEMPFLSH |                                           |
| 49 |                                                                                                                | TCF    | BANKJUMP |                                           |
| 50 |                                                                                                                |        |          |                                           |
| 51 | PINBRNCH                                                                                                       | RELINT |          | # FOR GOPIN USERS                         |
| 52 |                                                                                                                | CA     | MARK2PAC | # NEEDED TO SAVE MPAC +2 FOR MARK USERS   |
| 53 |                                                                                                                |        |          |                                           |
| 54 |                                                                                                                |        |          |                                           |
| 55 |                                                                                                                |        |          |                                           |
| 56 |                                                                                                                |        |          |                                           |
| 57 |                                                                                                                |        |          |                                           |
| 58 |                                                                                                                |        |          |                                           |
| 59 |                                                                                                                |        |          |                                           |
| 60 |                                                                                                                |        |          |                                           |

|          |       |          |    |                                            |
|----------|-------|----------|----|--------------------------------------------|
|          | TS    | MPAC     | +2 | # ONLY                                     |
|          | CA    | FLAGWRD4 |    | # PINBRANCH CONDITION                      |
|          | MASK  | PINMASK  |    |                                            |
|          | CCS   | A        |    |                                            |
|          | TCF   | +3       |    |                                            |
|          | TCF   | ERASER   |    | # ** NOTHING IN ENDIDLE                    |
|          | TCF   | MARKPLAY |    |                                            |
| NORMBNCH | TC    | UPFLAG   |    | # SET PINBRANCH BIT                        |
|          | ADRES | PINBRFLG |    |                                            |
|          | CAF   | PRIODBIT |    | # PRIO INTERRUPTED                         |
|          | MASK  | FLAGWRD4 |    |                                            |
|          | CCS   | A        |    |                                            |
|          | TCF   | KEEPPRIO |    |                                            |
|          | TCF   | PLAYJUM1 |    |                                            |
| NVDSP    | TC    | COPYPACS |    |                                            |
|          | CA    | TEMPOR2  |    | # SET UP BLANK BITS FOR NVMONOPT IN CASE   |
|          | MASK  | SEVEN    |    | # USER REQUESTS BLANKING MONITOR           |
|          | TS    | L        |    |                                            |
|          | CS    | BIT13    |    |                                            |
|          | INDEX | COPINDEX |    |                                            |
|          | MASK  | DSPFLG   |    |                                            |
|          | INDEX | COPINDEX |    |                                            |
|          | TS    | DSPFLG   |    |                                            |
|          | MASK  | BIT8     |    | # BIT8 SET IF DEC MARK PERFORM DISPLAY     |
|          | TS    | TEM1     |    |                                            |
|          | CA    | MPAC     | +2 |                                            |
|          | TS    | MPAC2SAV |    |                                            |
|          | TS    | MARK2PAC |    | # * FOR DISK ONLY *                        |
|          | INDEX | COPINDEX |    |                                            |
|          | CCS   | NVWORD   |    |                                            |
|          | TCF   | NVDSP1   |    |                                            |
|          | TCF   | CLEANEND |    |                                            |
|          | CS    | MARKNV   |    |                                            |
|          | TS    | MARKNV   |    | # IN CASE MARKPLAY AWAKENED AFTER SLEEPING |
|          | MASK  | LOW7     |    |                                            |
|          | AD    | V05N00M1 |    |                                            |
|          | AD    | TEM1     |    |                                            |
| NVDSP1   | AD    | ONE      |    |                                            |
| NV50DSP  | TC    | NVMONOPT |    |                                            |
|          | TCF   | REST     |    | # IF BUSY                                  |

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
|          | TC     | FLASHOFF | # IN CASE OF EXTENDED VERB NON-FLASH       |
|          | TC     | COPYTOGO | # MPACS DESTROYED BY NVSUB                 |
|          | TC     | DOWNFLAG | # UNSET SLEEPING BITS                      |
|          | ADRES  | MRKNVFLG |                                            |
|          | TC     | DOWNFLAG |                                            |
|          | ADRES  | NRMNVFLG |                                            |
|          | TC     | DOWNFLAG |                                            |
| BLANKCHK | ADRES  | PRONVFLG |                                            |
|          | CA     | TEMPOR2  | # BLANK BITS 1,2,3 IF SET                  |
|          | TC     | BLANKSUB |                                            |
| PERFCHEK | TCF    | NVDSP    |                                            |
|          | CAF    | BIT5     | # BIT5 FOR PERFORM                         |
|          | MASK   | TEMPOR2  |                                            |
|          | CCS    | A        | # IS THIS A GOPERF DISPLAY                 |
|          | TCF    | 1STOR2ND | # YES                                      |
| GOANIDLE | CAF    | BIT4     |                                            |
|          | MASK   | TEMPOR2  |                                            |
|          | CCS    | A        |                                            |
|          | TCF    | FLASHSUB | # IT IS                                    |
|          | CS     | TEMPOR2  | # IS THIS A GODSPRET                       |
|          | MASK   | BIT6     |                                            |
|          | CCS    | A        |                                            |
|          | TCF    | ISITN00  |                                            |
|          | INDEX  | COPINDEX |                                            |
|          | CA     | CADRFLSH |                                            |
|          | TS     | MPAC +3  |                                            |
|          | TCF    | ENDIT    |                                            |
| ISITN00  | INDEX  | COPINDEX | # IS THIS A PASTE                          |
|          | CA     | NVWORD   |                                            |
|          | MASK   | LOW7     | # CHECK MADE FOR PINBRNCH AND PRIO ON MARK |
|          | EXTEND |          |                                            |
|          | BZF    | FLASHSUB | # YES, ASSUME PASTE ALWAYS ON FLASH        |
|          | TCF    | ENDOFJOB | # NOT FLASH, NOT GOPERF, THEREFORE EXIT    |
| 1STOR2ND | CA     | TEMPOR2  |                                            |
|          | MASK   | BIT13    |                                            |
|          | CCS    | A        |                                            |
|          | TCF    | GOANIDLE | # SECOND                                   |
|          | CA     | BIT13    |                                            |
|          | INDEX  | COPINDEX |                                            |
|          | ADS    | DSPFLG   |                                            |
|          | ZL     |          |                                            |

|          |        |             |                                            |
|----------|--------|-------------|--------------------------------------------|
|          | EXTEND |             | # IS IT MARK                               |
|          | BZMF   | MARKPERF    | # YES                                      |
|          | MASK   | BIT12       |                                            |
|          | EXTEND |             |                                            |
|          | BZF    | V50PASTE    |                                            |
|          | CS     | NVWORD1     | # NVOWRD1= -0 IS V97. NVWORD1= -400 IS V99 |
|          | AD     | V97N00      |                                            |
| V50PASTE | TCF    | NV50DSP     |                                            |
|          | CAF    | V50N00      |                                            |
|          | TCF    | NV50DSP     | # DISPLAY SECOND PART OF GOPERF            |
| WITCHONE | CS     | BIT5        | # TURN OFF KEY RELEASE LIGHT               |
|          | EXTEND |             |                                            |
|          | WAND   | DSALMOUT    |                                            |
|          | CA     | FLAGWRD4    |                                            |
|          | MASK   | NVBUSMSK    | # IS IT NVSUB ALEEP                        |
|          | CCS    | A           |                                            |
|          | CAF    | ONE         |                                            |
|          | TS     | L           |                                            |
|          | CAF    | ZERO        |                                            |
|          | INDEX  | L           |                                            |
|          | XCH    | CADRSTOR    |                                            |
|          | INHINT |             |                                            |
|          | TC     | Q           |                                            |
| XCHTOEND | CAF    | ENDINST     | # TC ENDOFJOB REPLACES GENADR IN LOC FOR   |
| XCHNYLOC | XCH    | LOCCTR      | # WAS THIS ADDRESS SLEEPING                |
|          | EXTEND |             |                                            |
|          | BZMF   | RELINTQ     | # NO                                       |
|          | XCH    | LOCCTR      | # YES                                      |
|          | INDEX  | LOCCTR      |                                            |
|          | TS     | LOC         |                                            |
| RELINTQ  | RELINT |             |                                            |
|          | TC     | Q           | # BACK TO USER                             |
| CLEANEND | CAF    | PRI032      | # ONE LOWER THAN DISPLAYS SLEEPING         |
|          | TC     | FINDVAC     |                                            |
|          | EBANK= | NVSAVE      |                                            |
|          | 2CADR  | JAMTERM     |                                            |
|          | TCF    | FLASHSUB +1 |                                            |
| ISITPRIO | CA     | FLAGWRD4    |                                            |
|          | MASK   | ITISMASK    | # IS PINBRFLG, MARKIDFLG SET               |
|          | EXTEND |             |                                            |



|          |        |          |                                       |
|----------|--------|----------|---------------------------------------|
|          | BZF    | PRIOBORT |                                       |
|          | TCF    | ENDOFJOB |                                       |
| REST     | CCS    | CADRSTOR | # IS SOMEONE IN ENDIDLE               |
|          | TCF    | ENDOFJOB | # YES                                 |
|          | TCF    | RESTSLEP |                                       |
|          | TCF    | ENDOFJOB |                                       |
| RESTSLEP | CA     | GENMASK  | # SET NVSLEEP BITS                    |
|          | MASK   | ASTROMSK |                                       |
| OCT24100 | TC     | UPENT2   |                                       |
|          | OCT    | 24100    | # *** DON'T MOVE                      |
|          | INDEX  | COPINDEX |                                       |
|          | CAF    | NVCADR   |                                       |
|          | TC     | NVSUBUSY | # BUSY OR ABORT IF ILLEGAL            |
| FLASHSUB | TC     | FLASHON  |                                       |
|          | CA     | COPINDEX | # COPINDEX DESTROYED BY ENDIDLE       |
|          | TS     | COPMPAC  |                                       |
|          | CA     | GENMASK  |                                       |
|          | MASK   | IDLEMASK |                                       |
|          | TC     | UPENT2   |                                       |
| ITISMASK | OCT    | 40040    | # *** ENDIDLE ALLOW *** DON'T MOVE    |
|          | CA     | R1SAVE   | # IS THIS A REPEAT AND RETURN DISPLAY |
|          | INDEX  | COPINDEX |                                       |
|          | MASK   | BIT3     |                                       |
|          | CCS    | A        |                                       |
|          | TCF    | UNSETR1  | # YES                                 |
|          | CCS    | CADRSTOR | # SEE IF SOMEONE ALREADY IN ENDIDLE   |
|          | TCF    | ISITPRIO |                                       |
|          | TCF    | +2       |                                       |
|          | TCF    | ISITPRIO |                                       |
| IDLERET1 | TC     | ENDIDLE  |                                       |
|          | TCF    | TERMATE  |                                       |
|          | TCF    | PROCEED  | # ENDIDLE RETURNS HERE ON PROCEED     |
|          | CS     | LOWLOAD  |                                       |
|          | AD     | MPAC     | # VERBREG                             |
|          | EXTEND |          |                                       |
|          | DIM    | A        |                                       |
|          | EXTEND |          |                                       |
|          | BZF    | LOADITIS | # V21 OR V22 OR V23 ON DSKY           |

|                   |                   |                                 |                                                  |
|-------------------|-------------------|---------------------------------|--------------------------------------------------|
| OKTOENT<br>ENDOUT | CAF<br>TS         | TWO<br>OUTHERE                  |                                                  |
|                   | CA<br>MASK        | FLAGWRD4<br>OCT60000            | # CHECK NATURE OF ENDIDLE RETURN                 |
|                   | CCS<br>TCF<br>TCF | A<br>TIMECHEK<br>NORMRET        | # PRIO ENDIDLE RETURN<br># NORMAL ENDIDLE RETURN |
|                   | TCF               | MARKRET                         | # MARK ENDIDLE RETURN                            |
| TIMECHEK          | CS                | TIME1                           |                                                  |
|                   | AD<br>CCS<br>COM  | PRIOTIME<br>A                   |                                                  |
|                   | AD<br>AD<br>AD    | OCT37776<br>ONE<br>-2SEC        |                                                  |
|                   | EXTEND<br>BZMF    | KEEPPRIO                        |                                                  |
|                   | TCF               | NORMRET                         |                                                  |
| NORMWAKE          | CAF               | ONE                             |                                                  |
|                   | TCF               | WAKEPLAY                        |                                                  |
| ENDRET            | CCS               | OUTHERE                         |                                                  |
|                   | AD<br>TCF<br>TCF  | ONE<br>+2<br>ENDOFJOB           | # NORMAL ENDIDLE EXIT                            |
|                   | INDEX<br>AD<br>TS | COPMPAC<br>CADRFLSH<br>MPAC +3  |                                                  |
|                   | CA<br>MASK        | GENMASK<br>PINIDMSK             | # REMOVE ENDIDLE AND PINBRANCH BITS              |
| PINIDMSK          | TC<br>OCT         | DOWNENT2<br>74044               | # *** DON'T MOVE                                 |
|                   | CS<br>TC<br>TCF   | THREE<br>NVSUB<br>+1            | # BLANK EVERYTHING EXCEPT MM                     |
| ENDIT             | CA<br>MASK        | USERPRIO<br>PRIO37              | # RETURN TO USER'S PRIORITY                      |
|                   | TC<br>CA<br>TCF   | PRIOCHNG<br>MPAC +3<br>BANKJUMP |                                                  |
| UNSETR1           | INDEX<br>CS       | COPINDEX<br>BIT3                | # RESET REPEAT AND RETURN REQUEST                |

```
1
2 MASK R1SAVE
3 TS R1SAVE
4
5 CAF ZERO # *** 205 ONLY MARKBRAN USERS IN
6 TC SUPERSW # SUPERBANK 0
7
8 -1 CAF THREE # RETURN TO USER'S IMMEDIATE RETURN LOC
9 IMMEDRET INDEX COPINDEX
10
11 AD CADRFLSH
12 TCF BANKJUMP
13
14 TERMATE CAF ZERO # ASTRONAUT TERMINATE (V34) RETURNS TO
15 TCF ENDOUT
16
17 LINUSCHR CS PLAYTEM4 # IS THIS A LINUS
18 MASK BIT14
19 CCS A
20 TCF Q+1 # NO
21 CS PLAYTEM3 # YES, IS IT ALREADY IN ENDIDLE
22 INDEX COPINDEX
23 AD CADRFLSH
24 EXTEND BZF +2 # YES
25
26 TC Q # NO
27 CCS DSPLOCK # IS THE ASTRONAUT BUSY
28 TC ENDOFJOB # END THE NEW DISPLAY, IT'S ALREADY ACTIVE
29 TC Q
30
31 # MORE LOGIC COULD BE INCORPORATED HERE TO MAKE SURE A RECYCLE IS A RECYCLE AND CONVERSELY THAT A LOAD IS A LOAD
32
33 PROCEED CAF ONE # ASTRONAUT PROCEED (V33) RETURNS
34 TCF ENDOUT
35
36 # LASTPLAY CHECKS TO SEE IF (1) THE LAST NORMAL DISPLAY WAS EITHER INTERRUPTED BY A PRIO OR A MARK (MARK
37 # COULD ONLY HAPPEN DURING PINBRANCH) OR IF (2) THE LAST NORMAL DISPLAY WAS REQUESTED WHILE A HIGHER PRIORITY
38 # DISPLAY WAS GOING, RESULTING IN THE NORMAL BEING PUT TO SLEEP.
39 #
40 # IF EITHER OF THE ABOVE 2 CONDITIONS EXISTS, THE NORMAL DISPLAY IS AWAKENED TO GO TO PLAYJUM1 WHICH STARTS
41 # UP THE MOST RECENT VALID NORMAL DISPLAY. IF THESE 2 CONDITIONS DO NOT EXIST, CONTROL GOES TO PLAYJUM1 WHICH IS
42 # STARTED IMMEDIATELY WITH THE ASSUMPTION THAT THE MOST RECENT NORMAL DISPLAY IS ALREADY IN ENDIDLE (DURING A
43 # PINBRNCH) OR THAT A RESTART HAS OCCURRED AND THE DISPLAY CAN BE STARTED AS A .1 RESTART.
44
45 MARKRET CS SIX
46 MASK FLAGWRD4
47 INHINT TS FLAGWRD4 # *** MAY MOVE DISPLAY FLAGWORD OUT OF
48
49 RELINT TCF ENDRET # INHINT REALM
50
51
52
53
54
55
56
57
58
59
60
```

MARKOVER CAF MINUS1 # RUPTREG2 IS - MEANS ENDOFJOB TO ENDRET  
TS OUTHERE

CA FLAGWRD4 # IS ENDIDFLG SET  
MASK PRI030 # IS NORMAL OR PRIO IN ENDIDLE

CCS A  
TCF NORMBNCH

NORMRET CA FLAGWRD4 # IS MARK SLEEPING  
MASK BITS5+11 # OR WAITING

CCS A  
TCF MARKWAKE

CA FLAGWRD4 # NO  
MASK BITS4+10 # IS NORMAL INTERRUPTED OR WAITING  
CCS A  
TCF NORMWAKE # YES

CA EBANKTEM # NO, WAS IT A FLASH REQUEST  
MASK OCT50 # OR A GODSPRET

CCS A  
TCF ENDRET # YES  
CA NVSAVE

EXTEND  
BZF ENDRET

CAF PRI015  
INHINT  
TC NOVAC

EBANK= NVWORD  
2CADR PLAYJUM1

TCF ENDRET

MARSLEEP CA FLAGWRD4 # IS MARK ALREADY IN

MASK BITS5+11  
CCS A  
TCF ENDOFJOB # YES

TCF GOSLEEPS

LOADITIS INDEX COPMPAC

CA NVWORD  
MASK LOW7  
COM

AD MPAC +1 # NOUNREG

EXTEND  
BZF OKTOENT # NO, THEN LOAD IS VALID

TCF PINBRNCH # YES, ACCEPT LOAD BUT ASK FOR LAST AGAIN

|          |          |                |                                       |
|----------|----------|----------------|---------------------------------------|
| ERASER   | CS<br>TC | THREE<br>NVSUB | # BLANK EVERYTHING EXCEPT MM          |
|          | TCF      | ENDOFJOB       |                                       |
|          | TCF      | ENDOFJOB       |                                       |
| PERFMASK | OCT      | 0036           | # FLASH, PERFORM, BLANK R2 AND R3     |
| V01N25   | VN       | 00125          |                                       |
| V06N07   | VN       | 00607          | # GOPERF3 VN DISPLAY BEFORE V50       |
| V50N00   | VN       | 5000           |                                       |
| PERF2MSK | OCT      | 00030          | # FLASH, PERFORM                      |
| V04N06   | VN       | 00406          |                                       |
| PERF4MSK | OCT      | 14             | # FLASH, BLANK R3                     |
| GOAGIN   | EQUALS   | PINBRNCH       |                                       |
| REDOMASK | OCT      | 20010          | # BITS 4 AND 14                       |
| MARK3MSK | OCT      | 40230          | # MARK, DECIMAL NOUN, PERFORM, FLASH  |
| MARK4MSK | OCT      | 40036          | # MARK, PERFORM, FLASH, BLANK 2 AND 3 |
| NVCADR   | CADR     | REDOPRIO       |                                       |
| WAKECADR | CADR     | MARKPLAY       |                                       |
|          | CADR     | PLAYJUM1       |                                       |
| OCT3400  | OCT      | 3400           | # EBANK MASK                          |
| NBUSMASK | OCT      | 11210          |                                       |
| PMMASK   | OCT      | 66521          |                                       |
| VERBMASK | =        | MID7           | # (OCT 37600)                         |
| V05N00M1 | OCT      | 1177           | # V05 MINUS ONE                       |
| GOXDSP   | EQUALS   | GOMARK         |                                       |
| GOXDSPR  | EQUALS   | GOMARKR        |                                       |
| GOXDSPF  | EQUALS   | GOMARKF        |                                       |
| GOXDSPFR | EQUALS   | GOMARKFR       |                                       |
| ENEXT    | EQUALS   | ENDMARK        |                                       |
| MPAC2SAV | EQUALS   | BANKSET        |                                       |
| NVBUSMSK | OCT      | 700            |                                       |
| ASTROMSK | OCT      | 704            |                                       |
| MPERFMSK | OCT      | 40030          | # BIT 15,5,4 FOR MARK,PERFORM,FLASH   |
| OCT34300 | OCT      | 34300          |                                       |
| BITS15+7 | OCT      | 40100          |                                       |
| BITS7+4  | OCT      | 110            |                                       |
| DSPFLG   | EQUALS   | EBANKSAV       |                                       |
| MARKFLAG | EQUALS   | MARKEBAN       |                                       |
| SAVEFLAG | EQUALS   | EBANKTEM       |                                       |
| BITS5+11 | OCT      | 2020           | # * DON'T MOVE                        |
| BITS4+10 | OCT      | 1010           | # * DON'T MOVE                        |
| LOWLOAD  | DEC      | 22             |                                       |
| BUSYMASK | OCT      | 77730          |                                       |
| CADRMASK | OCT      | 50             |                                       |
| PINMASK  | EQUALS   | 13,14,15       |                                       |
| GOPLAY   | EQUALS   | NVDSP          |                                       |
| PRIOSAVE | EQUALS   | RISAVE         |                                       |
| COPMPAC  | EQUALS   | MPAC +3        |                                       |
| TEMPOR2  | EQUALS   | MPAC +4        |                                       |

[illegible]



# UPFLAG AND DOWNFLAG ARE ENTIRELY GENERAL FLAG SETTING AND CLEARING SUBROUTINES. USING THEM, WHETHER OR  
# NOT IN INTERRUPT, ONE MAY SET OR CLEAR ANY SINGLE, NAMED BIT IN ANY ERASABLE REGISTER, SUBJECT OF COURSE TO  
# EBANK SETTING. A "NAMED" BIT, AS THE WORD IS USED HERE, IS ANY BIT WITH A NAME FORMALLY ASSIGNED BY THE YUL  
# ASSEMBLER.

# AT PRESENT THE ONLY NAMED BITS ARE THOSE IN THE FLAGWORDS. ASSEMBLER CHANGES WILL MAKE IT POSSIBLE TO  
# NAME ANY BIT IN ERASABLE MEMORY.

# CALLING SEQUENCES ARE AS FOLLOWS :-

|   |       |              |       |              |
|---|-------|--------------|-------|--------------|
| # | TC    | UPFLAG       | TC    | DOWNFLAG     |
| # | ADRES | NAME OF FLAG | ADRES | NAME OF FLAG |

# RETURN IS TO THE LOCATION FOLLOWING THE "ADRES" ABOUT .58 MS AFTER THE "TC".  
# UPON RETURN A CONTAINS THE CURRENT FLAGWRD SETTING.

|        |           |
|--------|-----------|
| BLOCK  | 02        |
| SETLOC | FFTAG1    |
| BANK   |           |
| COUNT* | \$\$/FLAG |

|        |     |       |               |
|--------|-----|-------|---------------|
| UPFLAG | CA  | Q     |               |
|        | TC  | DEBIT |               |
|        | COM |       | # +(15 - BIT) |

|         |        |        |           |
|---------|--------|--------|-----------|
|         | EXTEND |        |           |
|         | ROR    | LCHAN  | # SET BIT |
| COMFLAG | INDEX  | ITEMP1 |           |

|        |          |
|--------|----------|
| TS     | FLAGWRD0 |
| LXCH   | ITEMP3   |
| RELINT |          |

|          |      |         |             |
|----------|------|---------|-------------|
|          | TC   | L       |             |
| DOWNFLAG | CA   | Q       |             |
|          | TC   | DEBIT   |             |
|          | MASK | L       | # RESET BIT |
|          | TCF  | COMFLAG |             |

|       |        |     |               |
|-------|--------|-----|---------------|
| DEBIT | AD     | ONE | # CET DE BITS |
|       | INHINT |     |               |

|    |        |
|----|--------|
| TS | ITEMP3 |
| CA | LOW4   |
| TS | ITEMP1 |

|       |        |
|-------|--------|
| INDEX | ITEMP3 |
| CA    | 0 -1   |
| TS    | L      |
| CA    | ZERO   |



|    |        |          |                               |    |
|----|--------|----------|-------------------------------|----|
| 1  |        |          |                               | 1  |
| 2  |        |          |                               | 2  |
| 3  | EXTEND |          |                               | 3  |
| 4  | DV     | ITEMP1   | # A = FLAGWRD, L = (15 - BIT) | 4  |
| 5  | DXCH   | ITEMP1   |                               | 5  |
| 6  | INDEX  | ITEMP1   |                               | 6  |
| 7  | CA     | FLAGWRD0 |                               | 7  |
| 8  | TS     | L        | # CURRENT STATE               | 8  |
| 9  | INDEX  | ITEMP2   |                               | 9  |
| 10 | CS     | BIT15    | # -(15 - BIT)                 | 10 |
| 11 | TC     | Q        |                               | 11 |
| 12 |        |          |                               | 12 |
| 13 |        |          |                               | 13 |
| 14 |        |          |                               | 14 |
| 15 |        |          |                               | 15 |
| 16 |        |          |                               | 16 |
| 17 |        |          |                               | 17 |
| 18 |        |          |                               | 18 |
| 19 |        |          |                               | 19 |
| 20 |        |          |                               | 20 |
| 21 |        |          |                               | 21 |
| 22 |        |          |                               | 22 |
| 23 |        |          |                               | 23 |
| 24 |        |          |                               | 24 |
| 25 |        |          |                               | 25 |
| 26 |        |          |                               | 26 |
| 27 |        |          |                               | 27 |
| 28 |        |          |                               | 28 |
| 29 |        |          |                               | 29 |
| 30 |        |          |                               | 30 |
| 31 |        |          |                               | 31 |
| 32 |        |          |                               | 32 |
| 33 |        |          |                               | 33 |
| 34 |        |          |                               | 34 |
| 35 |        |          |                               | 35 |
| 36 |        |          |                               | 36 |
| 37 |        |          |                               | 37 |
| 38 |        |          |                               | 38 |
| 39 |        |          |                               | 39 |
| 40 |        |          |                               | 40 |
| 41 |        |          |                               | 41 |
| 42 |        |          |                               | 42 |
| 43 |        |          |                               | 43 |
| 44 |        |          |                               | 44 |
| 45 |        |          |                               | 45 |
| 46 |        |          |                               | 46 |
| 47 |        |          |                               | 47 |
| 48 |        |          |                               | 48 |
| 49 |        |          |                               | 49 |
| 50 |        |          |                               | 50 |
| 51 |        |          |                               | 51 |
| 52 |        |          |                               | 52 |
| 53 |        |          |                               | 53 |
| 54 |        |          |                               | 54 |
| 55 |        |          |                               | 55 |
| 56 |        |          |                               | 56 |
| 57 |        |          |                               | 57 |
| 58 |        |          |                               | 58 |
| 59 |        |          |                               | 59 |
| 60 |        |          |                               | 60 |

# DELAYJOB- A GENERAL ROUTINE TO DELAY A JOB A SPECIFIC AMOUNT OF TIME BEFORE PICKING UP AGAIN.

#

# ENTRANCE REQUIREMENTS...

# CAF DT # DELAY JOB FOR DT CENTISECS

# TC BANKCALL

# CADR DELAYJOB

BANK 06

SETLOC DLAYJOB

BANK

# THIS MUST REMAIN IN BANK 0 \*\*\*\*\*

COUNT\* \$\$/DELAY

DELAYJOB INHINT

TS Q # STORE DELAY DT IN Q FOR DLY -1 IN

CAF DELAYNUM # WAITLIST

DELLOOP TS RUPTREG1

INDEX A

CA DELAYLOC # IS THIS DELAYLOC AVAILABLE

EXTEND

BZF OK2DELAY # YES

CCS RUPTREG1 # NO, TRY NEXT DELAYLOC

TCF DELLOOP

DXCH BUF2

TC BAILOUT1 # NO AVAILABLE LOCS.

OCT 1104

OK2DELAY CA TCSLEEP # SET WAITLIST IMMEDIATE RETURN

TS WAITEXIT

CA FBANK

AD RUPTREG1 # STORE BBANK FOR TASK CALL

TS L

CAF WAKECAD # STORE CADR FOR TASK CALL

TCF DLY2 -1 # DLY IS IN WAITLIST ROUTINE

TCGETCAD TC MAKECADR # GET CALLERS FCADR

INDEX RUPTREG1

TS DELAYLOC # SAVE DELAY CADRS

TC JOBSLEEP

WAKER CAF ZERO

INDEX BBANK

XCH DELAYLOC # MAKE DELAYLOC AVAILABLE



# SERVICE\_ROUTINES



|    |         |        |             |    |
|----|---------|--------|-------------|----|
| 1  |         |        |             | 1  |
| 2  |         | TC     | JOBWAKE     | 2  |
| 3  |         |        |             | 3  |
| 4  |         | TC     | TASKOVER    | 4  |
| 5  |         |        |             | 5  |
| 6  | TCSLEEP | GENADR | TCGETCAD -2 | 6  |
| 7  | WAKECAD | GENADR | WAKER       | 7  |
| 8  |         |        |             | 8  |
| 9  |         |        |             | 9  |
| 10 |         |        |             | 10 |
| 11 |         |        |             | 11 |
| 12 |         |        |             | 12 |
| 13 |         |        |             | 13 |
| 14 |         |        |             | 14 |
| 15 |         |        |             | 15 |
| 16 |         |        |             | 16 |
| 17 |         |        |             | 17 |
| 18 |         |        |             | 18 |
| 19 |         |        |             | 19 |
| 20 |         |        |             | 20 |
| 21 |         |        |             | 21 |
| 22 |         |        |             | 22 |
| 23 |         |        |             | 23 |
| 24 |         |        |             | 24 |
| 25 |         |        |             | 25 |
| 26 |         |        |             | 26 |
| 27 |         |        |             | 27 |
| 28 |         |        |             | 28 |
| 29 |         |        |             | 29 |
| 30 |         |        |             | 30 |
| 31 |         |        |             | 31 |
| 32 |         |        |             | 32 |
| 33 |         |        |             | 33 |
| 34 |         |        |             | 34 |
| 35 |         |        |             | 35 |
| 36 |         |        |             | 36 |
| 37 |         |        |             | 37 |
| 38 |         |        |             | 38 |
| 39 |         |        |             | 39 |
| 40 |         |        |             | 40 |
| 41 |         |        |             | 41 |
| 42 |         |        |             | 42 |
| 43 |         |        |             | 43 |
| 44 |         |        |             | 44 |
| 45 |         |        |             | 45 |
| 46 |         |        |             | 46 |
| 47 |         |        |             | 47 |
| 48 |         |        |             | 48 |
| 49 |         |        |             | 49 |
| 50 |         |        |             | 50 |
| 51 |         |        |             | 51 |
| 52 |         |        |             | 52 |
| 53 |         |        |             | 53 |
| 54 |         |        |             | 54 |
| 55 |         |        |             | 55 |
| 56 |         |        |             | 56 |
| 57 |         |        |             | 57 |
| 58 |         |        |             | 58 |
| 59 |         |        |             | 59 |
| 60 |         |        |             | 60 |

1412THE

# GENTRAN, A BLOCK TRANSFER ROUTINE.  
# WRITTEN BY D. EYLES

# MOD 1 BY KERNAN  
# MOD 2 BY SCHULENBERG (REMOVE RELINT)      SKIPPER REV 4 2/28/68

UTILITYM REV 17 11/18/67

#            THIS ROUTINE IS USEFULL FOR TRANSFERING N CONSECUTIVE ERASABLE OR FIXED QUANTITIES TO SOME OTHER N  
# CONSECUTIVE ERASABLE LOCATIONS. IF BOTH BLOCKS OF DATA ARE IN SWITCHABLE EBANKS, THEY MUST BE IN THE SAME ONE.

#            GENTRAN IS CALLABLE IN A JOB AS WELL AS A RUPT. THE CALLING SEQUENCE IS:

|   |      |       |         |                                        |
|---|------|-------|---------|----------------------------------------|
| # | I    | CA    | N-1     | # # OF QUANTITIES MINUS ONE.           |
| # | I +1 | TC    | GENTRAN | # IN FIXED-FIXED.                      |
| # | I +2 | ADRES | L       | # STARTING ADRES OF DATA TO BE MOVED.  |
| # | I +3 | ADRES | M       | # STARTING ADRES OF DUPLICATION BLOCK. |
| # | I +4 |       |         | # RETURNS HERE.                        |

#            GENTRAN TAKES 25 MCT'S (300 MICROSECONDS) PER ITEM + 5 MCT'S (60 MICS) FOR ENTERING AND EXITING.  
#            A, L, AND ITEMP1 ARE NOT PRESERVED.

BLOCK    02  
SETLOC   FFTAG4  
BANK

EBANK=   ITEMP1

COUNT\*   \$\$/TRAN

|         |        |            |                       |
|---------|--------|------------|-----------------------|
| GENTRAN | INHINT |            |                       |
|         | TS     | ITEMP1     | # SAVE N-1.           |
|         | INDEX  | Q          | # C(Q) = ADRES L.     |
|         | AD     | 0          | # ADRES (L + N - 1).  |
|         | INDEX  | A          |                       |
|         | CA     | 0          | # C(ABOVE).           |
|         | TS     | L          | # SAVE DATA.          |
|         | CA     | ITEMP1     |                       |
|         | INDEX  | Q          |                       |
|         | AD     | 1          | # ADRES (M + N - 1).  |
|         | INDEX  | A          |                       |
|         | LXCH   | 0          | # STUFF IT.           |
|         | CCS    | ITEMP1     | # LOOP UNTIL N-1 = 0. |
|         | TCF    | GENTRAN +1 |                       |
|         | TCF    | Q+2        | # RETURN TO CALLER.   |



|    |         |                                                                    |            |    |
|----|---------|--------------------------------------------------------------------|------------|----|
| 1  |         |                                                                    |            | 1  |
| 2  | # B5OFF | ZERO BIT 5 OF EXTVBACT, WHICH IS SET BY TESTXACT.                  |            | 2  |
| 3  | #       | MAY BE USED AS NEEDED BY ANY EXTENDED VERB WHICH HAS DONE TESTXACT |            | 3  |
| 4  |         |                                                                    |            | 4  |
| 5  |         | COUNT*                                                             | \$\$/EXTVB | 5  |
| 6  |         |                                                                    |            | 6  |
| 7  | B5OFF   | CS                                                                 | BIT5       | 7  |
| 8  |         | MASK                                                               | EXTVBACT   | 8  |
| 9  |         | TS                                                                 | EXTVBACT   | 9  |
| 10 |         | TC                                                                 | ENDOFJOB   | 10 |
| 11 |         |                                                                    |            | 11 |
| 12 |         |                                                                    |            | 12 |
| 13 |         |                                                                    |            | 13 |
| 14 |         |                                                                    |            | 14 |
| 15 |         |                                                                    |            | 15 |
| 16 |         |                                                                    |            | 16 |
| 17 |         |                                                                    |            | 17 |
| 18 |         |                                                                    |            | 18 |
| 19 |         |                                                                    |            | 19 |
| 20 |         |                                                                    |            | 20 |
| 21 |         |                                                                    |            | 21 |
| 22 |         |                                                                    |            | 22 |
| 23 |         |                                                                    |            | 23 |
| 24 |         |                                                                    |            | 24 |
| 25 |         |                                                                    |            | 25 |
| 26 |         |                                                                    |            | 26 |
| 27 |         |                                                                    |            | 27 |
| 28 |         |                                                                    |            | 28 |
| 29 |         |                                                                    |            | 29 |
| 30 |         |                                                                    |            | 30 |
| 31 |         |                                                                    |            | 31 |
| 32 |         |                                                                    |            | 32 |
| 33 |         |                                                                    |            | 33 |
| 34 |         |                                                                    |            | 34 |
| 35 |         |                                                                    |            | 35 |
| 36 |         |                                                                    |            | 36 |
| 37 |         |                                                                    |            | 37 |
| 38 |         |                                                                    |            | 38 |
| 39 |         |                                                                    |            | 39 |
| 40 |         |                                                                    |            | 40 |
| 41 |         |                                                                    |            | 41 |
| 42 |         |                                                                    |            | 42 |
| 43 |         |                                                                    |            | 43 |
| 44 |         |                                                                    |            | 44 |
| 45 |         |                                                                    |            | 45 |
| 46 |         |                                                                    |            | 46 |
| 47 |         |                                                                    |            | 47 |
| 48 |         |                                                                    |            | 48 |
| 49 |         |                                                                    |            | 49 |
| 50 |         |                                                                    |            | 50 |
| 51 |         |                                                                    |            | 51 |
| 52 |         |                                                                    |            | 52 |
| 53 |         |                                                                    |            | 53 |
| 54 |         |                                                                    |            | 54 |
| 55 |         |                                                                    |            | 55 |
| 56 |         |                                                                    |            | 56 |
| 57 |         |                                                                    |            | 57 |
| 58 |         |                                                                    |            | 58 |
| 59 |         |                                                                    |            | 59 |
| 60 |         |                                                                    |            | 60 |

# THE FOLLOWING SUBROUTINE MAY BE CALLED TO DISPLAY A NON-ABORTIVE ALARM CONDITION. IT MAY BE CALLED  
# EITHER IN INTERRUPT OR UNDER EXECUTIVE CONTROL.

#  
# CALLING SEQUENCE IS AS FOLLOWS:  
# TC ALARM  
# OCT AAANN # ALARM NO. NN IN GENERAL AREA AAA.  
# # (RETURNS HERE)

BLOCK 02  
SETLOC FFTAG7  
BANK

EBANK= FAILREG

COUNT\* \$\$/ALARM

# ALARM TURNS ON THE PROGRAM ALARM LIGHT, BUT DOES NOT DISPLAY.

ALARM INHINT

ALARM2 CA Q  
TS ALMCADR  
INDEX Q

BORTENT CA 0  
TS L

PRIDENT CA BBANK  
+1 EXTEND  
ROR SUPERBNK # ADD SUPER BITS.  
TS ALMCADR +1

LARMENT CA Q # STORE RETURN FOR ALARM  
TS ITEMP1

CHKFAIL1 CCS FAILREG # IS ANYTHING IN FAILREG  
TCF CHKFAIL2 # YES TRY NEXT REG  
LXCH FAILREG  
TCF PROGLARM # TURN ALARM LIGHT ON FOR FIRST ALARM

CHKFAIL2 CCS FAILREG +1  
TCF FAIL3  
LXCH FAILREG +1  
TCF MULTEXIT

FAIL3 CA FAILREG +2  
MASK POSMAX  
CCS A  
TCF MULTFAIL  
LXCH FAILREG +2  
TCF MULTEXIT

```
1
2
3 PROGLARM CS DSPTAB +11D
4
5 MASK OCT40400
6 ADS DSPTAB +11D
7
8 MULTEXIT XCH ITEMP1 # OBTAIN RETURN ADDRESS IN A
9 RELINT
10 INDEX A
11 TC 1
12
13 MULTFAIL CA L
14 AD BIT15
15 TS FAILREG +2
16
17 TCF MULTEXIT
18
19 # PRIOLARM DISPLAYS V05N09 VIA PRIODSPR WITH 3 RETURNS TO THE USER FROM THE ASTRONAUT AT CALL LOC +1,+2,+3 AND
20 # AN IMMEDIATE RETURN TO THE USER AT CALL LOC +4. EXAMPLE FOLLOWS,
21 #
22 # CAF OCTXX # ALARM CODE
23 # TC BANKCALL
24 # CADR PRIOLARM
25 # ...
26 # ...
27 # TC PHASCHNG # ASTRONAUT RETURN
28 # OCT X.1 # IMMEDIATE RETURN TO USER. RESTART
29 # # PHASE CHANGE FOR PRIO DISPLAY
30
31 BANK 10
32 SETLOC DISPLAYS
33 BANK
34
35 PRIOLARM COUNT* $$/DSPLA
36 INHINT
37 TS L # * * * * KEEP IN DISPLAY ROUTINES BANK
38 # SAVE ALARM CODE
39
40 CA BUF2
41 TS ALMCADR # 2 CADR OF PRIOLARM USER
42
43 CA BUF2 +1
44 TC PRIOENT +1 # * LEAVE L ALONE
45 -2SEC DEC -200 # *** DONT MOVE
46 CAF V05N09
47 TCF PRIODSPR
48
49 BLOCK 02
50 SETLOC FFTAG7
51 BANK
52
53 BAILOUT COUNT* $$/ALARM
54 INHINT
55 CA Q
```

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
|          | TS     | ALMCADR  |                                           |
|          | INDEX  | Q        |                                           |
|          | CAF    | 0        |                                           |
|          | TC     | BORTENT  |                                           |
| OCT40400 | OCT    | 40400    |                                           |
|          | INHINT |          |                                           |
| WHIMPER  | CA     | TWO      |                                           |
|          | AD     | Z        |                                           |
|          | TS     | BRUPT    |                                           |
|          | RESUME |          |                                           |
|          | TC     | POSTJUMP | # RESUME SENDS CONTROL HERE               |
|          | CADR   | ENEMA    |                                           |
| POOD00   | INHINT |          |                                           |
|          | CA     | Q        |                                           |
| ABORT2   | TS     | ALMCADR  |                                           |
|          | INDEX  | Q        |                                           |
|          | CAF    | 0        |                                           |
|          | TC     | BORTENT  |                                           |
| OCT77770 | OCT    | 77770    | # DON'T MOVE                              |
|          | CAF    | OCT35    | # 4.35SPOT FOR GOPOOD00                   |
|          | TS     | L        |                                           |
|          | COM    |          |                                           |
|          | DXCH   | -PHASE4  |                                           |
| GOPOOD00 | INHINT |          |                                           |
|          | TC     | BANKCALL | # RESET STATEFLG, REINTFLG, AND NODOFLAG. |
|          | CADR   | FLAGS    |                                           |
|          | CA     | FLAGWRD7 | # IS SERVICER CURRENTLY IN OPERATION?     |
|          | MASK   | V37FLBIT |                                           |
|          | CCS    | A        |                                           |
|          | TCF    | STRTIDLE |                                           |
|          | TC     | BANKCALL | # TERMINATE GRPS 1, 3, 5, AND 6           |
|          | CADR   | V37KLEAN |                                           |
|          | TC     | BANKCALL | # TERMINATE GRPS 2, 4, 1, 3, 5, AND 6     |
|          | CADR   | MR.KLEAN | # (I.E., GRP 4 LAST)                      |
|          | TCF    | WHIMPER  |                                           |
| STRTIDLE | CAF    | BBSERVDL |                                           |
|          | TC     | SUPERSW  |                                           |
|          | TC     | BANKCALL | # PUT SERVICER INTO ITS "GROUND" STATE    |
|          | CADR   | SERVIDLE | # AND PROCED TO GOTOPOOH.                 |
| CCSHOLE  | INHINT |          |                                           |
|          | CA     | Q        |                                           |
|          | TC     | ABORT2   |                                           |
| OCT1103  | OCT    | 1103     |                                           |
| CURTAINS | INHINT |          |                                           |
|          | CA     | Q        |                                           |
|          | TC     | ALARM2   |                                           |
| OCT217   | OCT    | 00217    |                                           |



|                                                              |        |          |                         |
|--------------------------------------------------------------|--------|----------|-------------------------|
|                                                              | TC     | ALMCADR  | # RETURN TO USER        |
| BAILOUT1                                                     | INHINT |          |                         |
|                                                              | DXCH   | ALMCADR  |                         |
|                                                              | CAF    | ADR40400 |                         |
| BOTHABRT                                                     | TS     | ITEMP1   |                         |
|                                                              | INDEX  | Q        |                         |
|                                                              | CAF    | 0        |                         |
|                                                              | TS     | L        |                         |
| POODOO1                                                      | TCF    | CHKFAIL1 |                         |
|                                                              | INHINT |          |                         |
|                                                              | DXCH   | ALMCADR  |                         |
|                                                              | CAF    | ADR77770 |                         |
|                                                              | TCF    | BOTHABRT |                         |
| ALARM1                                                       | INHINT |          |                         |
|                                                              | DXCH   | ALMCADR  |                         |
| ALMNCADR                                                     | INHINT |          |                         |
|                                                              | INDEX  | Q        |                         |
|                                                              | CA     | 0        |                         |
|                                                              | TS     | L        |                         |
|                                                              | TCF    | LARMENT  |                         |
| ADR77770                                                     | TCF    | OCT77770 |                         |
| ADR40400                                                     | TCF    | OCT40400 |                         |
| DOALARM                                                      | EQUALS | ENDOFJOB |                         |
|                                                              | EBANK= | DVCNTR   |                         |
| BBSERVDL                                                     | BBCON  | SERVIDLE |                         |
| # CALLING SEQUENCE FOR VARALARM                              |        |          |                         |
| #                                                            | CAF    | (ALARM)  |                         |
| #                                                            | TC     | VARALARM |                         |
| #                                                            |        |          |                         |
| # VARALARM TURNS ON PROGRAM ALARM LIGHT BUT DOES NOT DISPLAY |        |          |                         |
| VARALARM                                                     | INHINT |          |                         |
|                                                              | TS     | L        | # SAVE USERS ALARM CODE |
|                                                              | CA     | Q        | # SAVE USERS Q          |
|                                                              | TS     | ALMCADR  |                         |
|                                                              | TC     | PRIOENT  |                         |
| OCT14                                                        | OCT    | 14       | # DONT MOVE             |
|                                                              | TC     | ALMCADR  | # RETURN TO USER        |
| ABORT                                                        | EQUALS | WHIMPER  |                         |
|                                                              | BANK   | 13       |                         |
|                                                              | SETLOC | ABTFLGS  |                         |
|                                                              | BANK   |          |                         |

|    |       |        |            |    |
|----|-------|--------|------------|----|
| 1  |       |        |            | 1  |
| 2  |       | COUNT* | \$\$/ALARM | 2  |
| 3  |       |        |            | 3  |
| 4  | FLAGS | CS     | STATEBIT   | 4  |
| 5  |       | MASK   | FLAGWRD3   | 5  |
| 6  |       | TS     | FLAGWRD3   | 6  |
| 7  |       |        |            | 7  |
| 8  |       | CS     | REINTBIT   | 8  |
| 9  |       | MASK   | FLGWRD10   | 9  |
| 10 |       | TS     | FLGWRD10   | 10 |
| 11 |       |        |            | 11 |
| 12 |       | CS     | NODOBIT    | 12 |
| 13 |       | MASK   | FLAGWRD2   | 13 |
| 14 |       | TS     | FLAGWRD2   | 14 |
| 15 |       |        |            | 15 |
| 16 |       | TC     | Q          | 16 |
| 17 |       |        |            | 17 |
| 18 |       |        |            | 18 |
| 19 |       |        |            | 19 |
| 20 |       |        |            | 20 |
| 21 |       |        |            | 21 |
| 22 |       |        |            | 22 |
| 23 |       |        |            | 23 |
| 24 |       |        |            | 24 |
| 25 |       |        |            | 25 |
| 26 |       |        |            | 26 |
| 27 |       |        |            | 27 |
| 28 |       |        |            | 28 |
| 29 |       |        |            | 29 |
| 30 |       |        |            | 30 |
| 31 |       |        |            | 31 |
| 32 |       |        |            | 32 |
| 33 |       |        |            | 33 |
| 34 |       |        |            | 34 |
| 35 |       |        |            | 35 |
| 36 |       |        |            | 36 |
| 37 |       |        |            | 37 |
| 38 |       |        |            | 38 |
| 39 |       |        |            | 39 |
| 40 |       |        |            | 40 |
| 41 |       |        |            | 41 |
| 42 |       |        |            | 42 |
| 43 |       |        |            | 43 |
| 44 |       |        |            | 44 |
| 45 |       |        |            | 45 |
| 46 |       |        |            | 46 |
| 47 |       |        |            | 47 |
| 48 |       |        |            | 48 |
| 49 |       |        |            | 49 |
| 50 |       |        |            | 50 |
| 51 |       |        |            | 51 |
| 52 |       |        |            | 52 |
| 53 |       |        |            | 53 |
| 54 |       |        |            | 54 |
| 55 |       |        |            | 55 |
| 56 |       |        |            | 56 |
| 57 |       |        |            | 57 |
| 58 |       |        |            | 58 |
| 59 |       |        |            | 59 |
| 60 |       |        |            | 60 |

## # UPDATE\_PROGRAM

```
1 # PROGRAM NAME: P27
2 # WRITTEN BY: KILROY/ DE WOLF
3 #
4 # MOD NO: 6
5 # MOD BY: KILROY
6 # DATE: 01DEC67
7 #
8 # LOG SECTION: UPDATE PROGRAM.
9 #
10 # FUNCT. DESCR.: P27 (THE UPDATE PROGRAM) PROCESSES COMMANDS AND DATA
11 # INSERTIONS REQUESTED BY THE GROUND VIA UPLINK.
12 #
13 # THE P27 PROGRAM WILL ACCEPT UPDATES
14 # ONLY DURING P00 FOR THE LM, AND ONLY DURING P00,
15 # P02, AND FRESH START FOR THE CSM
16 #
17 # CALLING SEQ: PROGRAM IS INITIATED BY UPLINK ENTRY OF VERBS 70, 71, 72, AND 73.
18 #
19 # SUBROUTINES: TESTXACT, NEWMODEX, NEWMODEX +3, GOXDSPF, BANKCALL, FINDVAC, INTPRET, INTSTALL, TPAGREE,
20 # INTWAKEU, ENDEXT, POSTJUMP, FALTON, NEWPHASE, PHASCHNG
21 #
22 # NORMAL EXIT: TC ENDEXT
23 #
24 # ALARM/ABORT: TC FALTON FOLLOWED BY TC ENDEXT
25 #
26 # RESTARTS: P27 IS RESTART PROTECTED IN TWO WAYS ...
27 # 1. PRIOR TO VERIFLAG INVERSION (WHICH IS CAUSED BY THE GROUND/ASTRONAUT'S VERIFICATION OF UPDATE
28 # DATA BY SENDING A V33E WHEN V21N02 IS FLASHING)---
29 # NO PROTECTION EXCEPT PRE-P27 MODE IS RESTORED, COAST + ALIGN DOWNLIST IS SELECTED AND UPLINK
30 # ACTIVITY LIGHT IS TURNED OFF. (JUST AS IF A V34E WAS SENT DURING P27 DATA LOADS).
31 # V70,V71,V72, OR V73 WILL HAVE TO BE COMPLETELY RESENT BY USER.
32 # 2. AFTER VERIFLAG INVERSION (WHEN UPDATE OF THE SPECIFIED ERASABLES IS BEING PERFORMED)---
33 # PROTECTED AGAINST RESTARTS.
34 #
35 # DEBRIS: UPBUFF (20D) TEMP STORAGE FOR ADDRESSES AND CONTENTS.
36 # UPVERB (1) VERB NUMBER MINUS 70D (E.G., FOR V72, UPVERB = 72D - 70D = 2)
37 # UPOLDMOD(1) FOR MAJOR MODE INTERRUPTED BY P27.
38 # COMPNUMB(1) TOTAL NUMBER OF COMPONENTS TO BE TRANSMITTED.
39 # UPCOUNT (1) ACTUAL NUMBER OF COMPONENTS RECEIVED.
40 # UPTEMP (1) SCRATCH, BUT USUALLY CONTAINS COMPONENT NUMBER TO BE CHANGED DURING VERIFY CYCLE
41 #
42 # INPUT:
43 #
44 # ENTRY DESCRIPTION
45 # V70EXXXXXXEXXXXXE (LIFTOFF TIME INCREMENT) DOUBLE PRECISION OCTAL TIME INCREMENT, XXXXX XXXXX,
46 # IS ADDED TO TEPHEM, SUBTRACTED FROM AGC CLOCK(TIME2,TIME1), SUBTRACTED FROM CSM STATE
47 # VECTOR TIME(TETCSM) AND SUBTRACTED FROM LEM STATE VECTOR TIME(TETLEM).
48 # THE DP OCTAL TIME INCREMENT IS SCALED AT 2(28).
```

```
1 # V71EIIIEAAAAE (CONTIGUOUS BLOCK UPDATE) II-2 OCTAL COMPONENTS, XXXXX,
2 # XXXXXE ARE LOADED INTO ERASABLE STARTING AT ECADR, AAAA.
3 # XXXXXE IT IS .GE. 3 .AND. .LE. 200.,
4 # ... AND (AAAA + II -3) DOES NOT PRODUCE AN ADDRESS IN THE
5 # NEXT BANK.
6 # SCALING IS SAME AS INTERNAL REGISTERS.
7 # V72EIIIE (SCATTER UPDATE) (II-1)/2 OCTAL COMPONENTS, XXXXX, ARE
8 # AAAAEXXXXXE LOADED INTO ERASABLE LOCATIONS, AAAA.
9 # AAAAEXXXXXE II IS .GE. 3 .AND. .LE. 19D, AND MUST BE ODD.
10 # ... SCALING IS SAME AS INTERNAL REGISTERS.
11 #
12 # V73EXXXXXEXXXXXE (OCTAL CLOCK INCREMENT) DOUBLE PRECISION OCTAL TIME
13 # INCREMENT XXXXX XXXXX, IS ADDED TO THE AGC CLOCK, IN
14 # CENTISECONDS SCALED AT (2)28).
15 # THIS LOAD IS THE OCTAL EQUIVALENT OF V55.
16 #
17 #
18 # OUTPUT: IN ADDITION TO THE ABOVE REGISTER LOADS, ALL UPDATES
19 # COMPLEMENT BIT3 OF FLAGWORD7.
20 #
21 # ADDITIONAL NOTES: VERB 71, JUST DEFINED ABOVE WILL BE USED TO PERFORM BUT NOT LIMITED TO THE FOLLOWING UPDATES --
22 # 1. CSM/LM STATE VECTOR UPDATE
23 # 2. REFSMMAT UPDATE
24 #
25 # THE FOLLOWING COMMENTS DELINEATE EACH SPECIAL UPDATE ---
26 #
27 # 1. CSM/LM STATE VECTOR UPDATE (ALL DATA ENTRIES IN OCTAL)
28 # ENTRIES: DATA DEFINITION: SCALE FACTORS:
29 # V71E CONTIGUOUS BLOCK UPDATE VERB
30 # 21E NUMBER OF COMPONENTS FOR STATE VECTOR UPDATE
31 # AAAAE ECADR OF 'UPSVFLAG'
32 # XXXXXE STATE VECTOR IDENTIFIER: 00001 FOR CSM, 77776 FOR LEM -- EARTH SPHERE OF INFLUENCE SCALING
33 # 00002 FOR CSM, 77775 FOR LEM -- LUNAR SPHERE OF INFLUENCE SCALING
34 # XXXXXEXXXXXE X POSITION
35 # XXXXXEXXXXXE Y POSITION
36 # XXXXXEXXXXXE Z POSITION
37 # XXXXXEXXXXXE X VELOCITY
38 # XXXXXEXXXXXE Y VELOCITY
39 # XXXXXEXXXXXE Z VELOCITY
40 # XXXXXEXXXXXE TIME FROM AGC CLOCK ZERO
41 # V33E VERB 33 TO SIGNAL THAT THE STATE VECTOR IS READY TO BE STORED.
42 #
43 # 2. REFSMMAT (ALL DATA ENTRIES IN OCTAL)
44 # ENTRIES: DATA DEFINITIONS: SCALE FACTORS:
```

```
1 # V71E CONTIGUOUS BLOCK UPDATE VERB
2 # 24E NUMBER OF COMPONENTS FOR REFSMMAT UPDATE
3 # AAAAE ECADR OF `REFSMMAT'
4 # XXXXXE XXXXE ROW 1 COLUMN 1 2(-1)
5 # XXXXXE XXXXE ROW 1 COLUMN 2 2(-1)
6 # XXXXXE XXXXE ROW 1 COLUMN 3 2(-1)
7 # XXXXXE XXXXE ROW 2 COLUMN 1 2(-1)
8 # XXXXXE XXXXE ROW 2 COLUMN 2 2(-1)
9 # XXXXXE XXXXE ROW 2 COLUMN 3 2(-1)
10 # XXXXXE XXXXE ROW 3 COLUMN 1 2(-1)
11 # XXXXXE XXXXE ROW 3 COLUMN 2 2(-1)
12 # XXXXXE XXXXE ROW 3 COLUMN 3 2(-1)
13 # V33E VERB 33 TO SIGNAL THAT REFSMMAT IS READY TO BE STORED
14
15
16
17
18 BANK 07
19 SETLOC EXTVERBS
20 BANK
21
22 EBANK= TEPHEM
23
24 COUNT* $$/P27
25 V70UPDAT CAF UP70 # COMES HERE ON V70E
26 TCF V73UPDAT +1
27
28 V71UPDAT CAF UP71 # COMES HERE ON V71E
29 TCF V73UPDAT +1
30
31 V72UPDAT CAF UP72 # COMES HERE ON V72E
32 TCF V73UPDAT +1
33
34 V73UPDAT CAF UP73 # COMES HERE ON V73E
35
36 +1 TS UPVERBSV # SAVE UPVERB UNTIL IT'S OK TO ENTER P27
37
38 TC TESTXACT # GRAB DISPLAY IF AVAILABLE, OTHERWISE
39 # TURN*OPERATOR ERROR* ON AND TERMINATEJOB
40
41 CA MODREG # CHECK IF UPDATE ALLOWED
42 EXTEND # FIRST CHECK FOR MODREG = +0, -0
43 BZF +3 # (+0 = P00, -0 = FRESHSTART)
44 UPERROR TC POSTJUMP # TURN ON 'OPERATOR ERROR' LIGHT
45 CADR UPERROUT +2 # GO TO COMMON UPDATE PROGRAM EXIT
46
47 CAE MODREG # UPDATE ALLOWED.
48 CKMDMORE = UPERROR
49 TS UPOLDMOD # SAVE CURRENT MAJOR MODE
50
51
52
53
54
55
56
57
58
59
60
```

```
1
2 CAE UPVERBSV # SET UPVERB TO INDICATE TO P27
3
4 TS UPVERB # WHICH EXTENDED VERB CALLED IT.
5
6 CAF ONE
7 TS UPCOUNT # INITIALIZE UPCOUNT TO 1
8
9 TC POSTJUMP # LEAVE EXTENDED VERB BANK AND
10 CADR UPPART2 # GO TO UPDATE PROGRAM (P27) BANK.
11
12 UP70 EQUALS ZERO
13 UP71 EQUALS ONE
14 UP72 EQUALS TWO
15 UP73 EQUALS THREE
16
17 BANK 04
18 SETLOC UPDATE2
19 BANK
20
21 COUNT* $$/P27
22
23 UPPART2 EQUALS # UPDATE PROGRAM -- PART 2
24
25 TC PHASCHNG # SET RESTART GROUP 6 TO RESTORE OLD MODE
26 OCT 07026 # AND DOWNLIST AND EXIT IF RESTART OCCURS.
27 OCT 30000 # PRIORITY SAME AS CHRPRIO
28
29 SBANK= PINSUPER
30 EBANK= UPBUFF
31 2CADR UPOUT +1
32
33 CAF ONE
34 TS DNLSTCOD # DOWNLIST
35
36 TC NEWMODEX # SET MAJOR MODE = 27
37 DEC 27
38
39 INDEX UPVERB # BRANCH DEPENDING ON WHETHER THE UPDATE
40 TCF +1 # VERB REQUIRES A FIXED OR VARIABLE NUMBER
41 TCF +3 # V70 FIXED (OF COMPONENTS)
42 TCF OHWELL1 # V71 VARIABLE -- GO GET NO. OF COMPONENTS
43 TCF OHWELL1 # V72 VARIABLE -- GO GET NO. OF COMPONENTS
44 CA TWO # V73 (AND V70) FIXED
45 TS COMPNUMB # SET NUMBER OF COMPONENTS TO 2.
46 TCF OHWELL2 # GO GET THE TWO UPDATE COMPONENTS
47
48 OHWELL1 CAF ADUPBUFF # * REQUEST USER TO SEND NUMBER *
49 TS MPAC +2 # * OF COMPONENTS PARAMETER(II).*
50 +2 CAF UPLOADNV # (CKV432 RETURNS HERE IF V32 ENCOUNTERED)
51 TC BANKCALL # DISPLAY A FLASHING V21N01
52
53
54
55
56
57
58
59
60
```

```
1
2 CADR GOXDSPF # TO REQUEST II.
3
4 TCF UPOUT4 # V34 TERMINATE UPDATE (P27) RETURN
5 TCF OHWELL1 +2
6 TC CK4V32 # DATA OR V32 RETURN
7
8 CS BIT2
9 AD UPBUFF # IS II (NUMBER OF COMPONENTS PARAMETER)
10 EXTEND # .GE. 3 AND .LE. 20D.
11
12 BZMF OHWELL1 +2
13 CS UPBUFF
14 AD UP21
15
16 EXTEND
17 BZMF OHWELL1 +2
18 CAE UPBUFF
19
20 TS COMPNUMB # SAVE II IN COMPNUMB
```

## # UPBUFF LOADING SEQUENCE

```
21 OHWELL2 INCR UPCOUNT # INCREMENT COUNT OF COMPONENTS RECEIVED.
22 CAF ADUPBFM1 # CALCULATE LOCATION (ECADR) IN UPBUFF
23
24 +2 AD UPCOUNT # WHERE NEXT COMPONENT SHOULD BE STORED
25 +3 TS MPAC +2 # PLACE ECADR INTO R3.
26
27 CAF UPLOADNV # (CK4V32 RETURNS HERE IF V32 ENCOUNTERED)
28
29 TC BANKCALL # DISPLAY A FLASHING V21N01
30 CADR GOXDSPF # TO REQUEST DATA.
31 TCF UPOUT4 # V34 TERMINATE UPDATE (P27) RETURN.
32
33 TCF OHWELL2 +3 # V33 PROCEED RETURN
34 TC CK4V32 # DATA OR V32 RETURN
35 CS UPCOUNT # HAVE WE FINISHED RECEIVING ALL
36
37 AD COMPNUMB # THE DATA WE EXPECTED.
38 EXTEND
39 BZMF UPVERIFY # YES -- GO TO VERIFICATION SEQUENCE
40 TCF OHWELL2 -1 # NO -- REQUEST ADDITIONAL DATA.
```

## # VERIFY SEQUENCE

```
41 UPVERIFY CAF ADUPTMP # PLACE ECADR WHERE COMPONENT NO. INDEX
42 TS MPAC +2 # IS TO BE STORED INTO R3.
43
44 CAF UPVRFYNV # (CK4V32 RETURNS HERE IF V32 ENCOUNTERED)
45 TC BANKCALL # DISPLAY A FLASHING V21N02 TO REQUEST
46 CADR GOXDSPF # DATA CORRECTION OR VERIFICATION.
47
48 TCF UPOUT4 # V34 TERMINATE UPDATE (P27) RETURN
49 TCF UPSTORE # V33 DATA SENT IS GOOD. GO STORE IT.
50 TC CK4V32 # COMPONENT NO. INDEX OR V32 RETURN
51
52 CA UPTMP # DOES THE COMPONENT NO. INDEX JUST SENT
53 EXTEND # SPECIFY A LEGAL COMPONENT NUMBER?
54 BZMF UPVERIFY # NO, IT IS NOT POSITIVE NONZERO
55
56 CS UPTMP
57 AD COMPNUMB
```

```
1
2 AD BIT1
3 EXTEND
4 BZMF UPVERIFY # NO
5 CAF ADUPBFM1 # YES -- BASED ON THE COMPONENT NO. INDEX
6 AD UPTMP # CALCULATE THE ECADR OF LOCATION IN
7 TCF OHWELL2 +2 # UPBUFF WHICH USER WANTS TO CHANGE.
8
9 UPOUT4 EQUALS UPOUT +1 # COMES HERE ON V34 TO TERMINATE UPDATE
10
11 # CHECK FOR VERB 32 SEQUENCE
12
13 CK4V32 CS MPAC # ON DATA RETURN FROM `GOXDSPF'
14 MASK BIT6 # ON DATA RETURN FROM "GOXDSP" & THE CON-
15 CCS A # TENTS OF MPAC = VERB. SO TEST FOR V32.
16 TC Q # IT'S NOT A V32, IT'S DATA. PROCEED.
17 INDEX Q
18 TC 0 -6 # V32 ENCOUNTERED -- GO BACK AND GET DATA
19
20 ADUPTMP ADRES UPTMP # ADDRESS OF TEMP STORAGE FOR CORRECTIONS
21 ADUPBUFF ADRES UPBUFF # ADDRESS OF UPDATE DATA STORAGE BUFFER
22 UPLOADNV VN 2101 # VERB 21 NOUN 01
23 UPVRFYNV VN 2102 # VERB 21 NOUN 02
24 UP21 = MD1 # DEC 21 = MAX NO OF COMPONENTS +1
25 UPDTPHAS EQUALS FIVE
26
27 # PRE-STORE AND FAN TO APPROPRIATE BRANCH SEQUENCE
28
29 UPSTORE EQUALS # GROUND HAS VERIFIED UPDATE. STORE DATA.
30
31 INHINT
32
33 CAE FLAGWRD7 # INVERT VERIFLAG (BIT 3 OF FLAGWRD7) TO
34 XCH L # INDICATE TO THE GROUND (VIA DOWNLINK)
35 CAF VERIFBIT # THAT THE V33 (WHICH THE GROUND SENT TO
36 EXTEND # VERIFY THE UPDATE) HAS BEEN SUCCESSFULLY
37 RXOR LCHAN
38 TS FLAGWRD7 # RECEIVED BY THE UPDATE PROGRAM
39
40 TC PHASCHNG # SET RESTART GROUP 6 TO REDO THE UPDATE
41 OCT 04026 # DATA STORE IF A RESTART OCCURS.
42 INHINT # (BECAUSE PHASCHNG DID A RELINT)
43
44 CS TWO # GO TO UPFNDVAC IF INSTALL IS REQUIRED.
45 AD UPVERB # THAT IS, IF IT'S A V70 - V72.
46 EXTEND # GO TO UPEND73 IF IT'S A V73.
47 BZMF UPFNDVAC
48
49 # VERB 73 BRANCH
```



```
1 UPEND73 EXTEND # V73 -- PERFORM DP OCTAL AGC CLOCK INCREMENT
2
3
4 DCA UPBUFF
5 DXCH UPBUFF +8D
6 TC TIMEDIDL
7 TC FALTON # ERROR -- TURN ON *OPERATOR ERROR* LIGHT
8 TC UPOUT +1 # GO TO COMMON UPDATE PROGRAM EXIT
9
10 UPFNDVAC CAF CHRPRIO # (USE EXTENDED VERB PRIORITY)
11 TC FINDVAC # GET VAC AREA FOR 'CALL INTSTALL'
12 EBANK= TEPHEM
13 2CADR UPJOB # (NOTE: THIS WILL ALSO SET EBANK FOR
14 TC ENDOFJOB # 'TEPHEM' UPDATE BY V70)
15
16 UPJOB TC INTPRET # THIS COULD BE A STATE VECTOR UPDATE -- SO
17 CALL INTSTALL # WAIT (PUT JOB TO SLEEP) IF ORBIT INT(OI)
18 # IS IN PROGRESS -- OR -- GRAB OI AND RETURN
19 # TO UPWAKE IF OI IS NOT IN PROGRESS.
20
21 UPWAKE EXIT
22
23 TC PHASCHNG # RESTART PROTECT (GROUP 6)
24 OCT 04026
25
26 TC UPFLAG # SET INTEGRATION RESTART BIT
27 ADRES REINTFLG
28
29 UPPART3 INHINT
30 EQUALS
31
32 INDEX UPVERB # BRANCH TO THE APPROPRIATE UPDATE VERB
33 TCF +1 # ROUTINE TO ACTUALLY PERFORM THE UPDATE
34 TCF UPEND70 # V70
35 TCF UPEND71 # V71
36 TCF UPEND72 # V72
37
38 # ROUTINE TO INCREMENT CLOCK (TIME2,TIME1) WITH CONTENTS OF DP WORD AT UPBUFF.
39
40 TIMEDIDL EXTEND
41 QXCH UPTMP # SAVE Q FOR RETURN
42 CAF ZERO # ZERO AND SAVE TIME2,TIME1
43 ZL
44 DXCH TIME2
45 DXCH UPBUFF +18D # STORE IN CASE OF OVERFLOW
46
47 CAF UPDTPHAS # SO
48 TS L # A
49 COM # QUICK
50 DXCH -PHASE6 # PHASCHNG
51
52
53
54
55
56
57
58
59
60
```

TIMEDIDR

INHINT

CAF ZERO

ZL

TS

MPAC +2

DXCH

UPBUFF +8D

DXCH

MPAC

# PICK UP INCRMENTER (AND ZERO

# IT IN CASE OF RESTARTS) AND

# STORE IT

# INTO MPAC FOR TPAGREE.

EXTEND

DCA

UPBUFF +18D

DAS

MPAC

# FORM SUM IN MPAC

EXTEND

BZF

DELTAOK

CAF

ZERO

# TEST FOR OVERFLOW

DXCH

UPBUFF +18D

DAS

TIME2

# OVERFLOW, RESTORE OLD VALUE OF CLOCK

# AND TURN ON OPERATOR ERROR

TC

PHASCHNG

OCT

04026

# RESTART PROTECT (GROUP 6)

TC

UPTMP

# GO TO ERROR EXIT

DELTAOK

TC

TPAGREE

# FORCE SIGN AGREEMENT

DXCH

MPAC

DAS

TIME2

# INCREMENT TIME2,TIME1

TC

PHASCHNG

OCT

04026

# RESTART PROTECT (GROUP 6)

INHINT

INDEX

UPTMP

TC

1

# (CODED THIS WAY FOR RESTART PROTECTION)

# NORMAL RETURN

# VERB 71 BRANCH

UPEND71

CAE

UPBUFF +1

TS

EBANK

MASK

LOW8

TS

UPTMP

# SET EBANK

# AND

# CALCULATE

# S-REG VALUE OF RECEIVING AREA

AD

NEG3

AD

COMPNUMB

EXTEND

BZF

STORLP71

MASK

BIT9

CCS

A

TCF

UPERROUT

# IN THE PROCESS OF

# PERFORMING

# THIS UPDATE

# WILL WE

# OVERFLOW

# INTO THE NEXT EBANK....

# YES

CA

NEG3

AD

COMPNUMB

TS

MPAC

# NO -- CALCULATE NUMBER OF

# WORDS TO BE STORED MINUS ONE

STORLP71

# SAVE NO. OF WORDS REMAINING MINUS ONE

```
1
2 INDEX A # TAKE NEXT UPDATE WORD FROM
3 CA UPBUFF +2 # UPBUFF AND
4
5 TS L # SAVE IT IN L
6 CA MPAC # CALCULATE NEXT
7
8 AD UPTMP # RECEIVING ADDRESS
9 INDEX A
10 EBANK= 1400
11
12 LXCH 1400 # UPDATE THE REGISTER BY CONTENTS OF L
13 EBANK= TEPHEM
14 CCS MPAC # ARE THERE ANY WORDS LEFT TO BE STORED
15
16 TCF STORLP71 # YES
17 TCF UPOUT # NO -- THEN EXIT UPDATE PROGRAM
18 ADUPBFM1 ADRES UPBUFF -1 # SAME AS ADUPBUFF BUT LESS 1 (DON'T MOVE)
19
20 TCF UPOUT # NO -- EXIT UPDATE (HERE WHEN COMPNUMB = 3)
21
22 # VERB 72 BRANCH
23
24 UPEND72 CAF BIT1 # HAVE AN ODD NO. OF COMPONENTS
25 MASK COMPNUMB # BEEN SENT FOR A V72 UPDATE ...
26
27 CCS A
28 TCF +2 # YES
29 TCF UPERROUT # ERROR -- SHOULD BE ODD NO. OF COMPONENTS
30
31 LDLOOP72 CS BIT2
32 AD COMPNUMB
33 TS MPAC # NOW PERFORM THE UPDATE
34
35 INDEX A
36 CAE UPBUFF +1 # PICK UP NEXT UPDATE WORD
37 LXCH A
38
39 CCS MPAC # SET POINTER TO ECADR (MUST BE CCS)
40 TS MPAC
41 INDEX A
42
43 CAE UPBUFF +1 # PICK UP NEXT ECADR OF REG TO BE UPDATED
44 TS EBANK # SET EBANK
45 MASK LOW8 # ISOLATE RELATIVE ADDRESS
46
47 INDEX A
48 EBANK= 1400
49 LXCH 1400 # UPDATE THE REGISTER BY CONTENTS OF L
50
51 EBANK= TEPHEM
52 CCS MPAC # ARE WE THORUGH THE V72 UPDATE...
53 TCF LDLOOP72 # NO
54
55 # NORMAL FINISH OF P27
56
57 UPOUT EQUALS
58 TC INTWAKEU # RELEASE GRAB OF ORBITAL INTEGRATION
59 +1 CAE UPOLDMOD # RESTORE PRIOR P27 MODE
60
61 TC NEWMODEX +3
62 CAF ZERO
```

```
1
2 TS DNLSTCOD
3 TC UPACTOFF # TURN OFF 'UPLINK ACTIVITY' LIGHT
4
5 EXTEND
6 DCA NEG0 # KILL GROUP 6.
7 DXCH -PHASE6
8
9 TC ENDEXT # EXTENDED VERB EXIT
10
11 # VERB TO BRANCH
12
13 UPEND70 EXTEND # V70 DOES THE FOLLOWING WITH DP DELTA
14 DCS UPBUFF # TIME IN UPBUFF
15 DXCH UPBUFF +8D
16 TC TIMEDIDL # DECREMENT AGC CLOCK
17
18 TC UPERROUT # ERROR WHILE DECREMENTING CLOCK -- EXIT
19
20 EBANK= TEPHEM
21 EXTEND
22 DCS UPBUFF # COPY DECREMENTERS FOR
23 DXCH UPBUFF +10D # RESTART PROTECTION
24 EXTEND
25 DCS UPBUFF
26 DXCH UPBUFF +12D
27
28 TC PHASCHNG # RESTART PROTECT (GROUP 6)
29 OCT 04026
30
31 CAF ZERO
32 ZL
33 DXCH UPBUFF +10D # DECREMENT CSM STATE VECTOR TIME
34 DAS TETCSM
35
36 CAF ZERO
37 ZL
38 DXCH UPBUFF +12D # DECREMENT LEM STATE VECTOR TIME
39 DAS TETLEM
40
41 CAF ZERO
42 ZL
43 DXCH UPBUFF
44 DAS TEPHEM +1 # INCREMENT TP TEPHEM
45 ADS TEPHEM
46
47 TC PHASCHNG # RESTART PROTECT (GROUP 6)
48 OCT 04026
49
50 EBANK= UPBUFF
51
52
53
54
55
56
57
58
59
60
```



# UPDATE\_PROGRAM

|    |                                                                                                     |        |          |                                           |    |
|----|-----------------------------------------------------------------------------------------------------|--------|----------|-------------------------------------------|----|
| 1  |                                                                                                     |        |          |                                           | 2  |
| 2  |                                                                                                     | TC     | UPOUT    | # GO TO STANDARD UPDATE PROGRAM EXIT      | 3  |
| 3  |                                                                                                     |        |          |                                           | 4  |
| 4  |                                                                                                     |        |          |                                           | 5  |
| 5  | # ERROR SEQUENCE                                                                                    |        |          |                                           | 6  |
| 6  |                                                                                                     |        |          |                                           | 7  |
| 7  | UPERROUT                                                                                            | TC     | FALTON   | # TURN ON *OPERATOR ERROR* LIGHT          | 8  |
| 8  |                                                                                                     | TCF    | UPOUT    | # GO TO COMMON UPDATE PROGRAM EXIT        | 9  |
| 9  |                                                                                                     |        |          |                                           | 10 |
| 10 | +2                                                                                                  | TC     | FALTON   | # TURN ON 'OPERATOR ERROR' LIGHT          | 11 |
| 11 |                                                                                                     | TC     | UPACTOFF | # TURN OFF 'UPLINK ACTIVITY' LIGHT        | 12 |
| 12 |                                                                                                     | TC     | ENDEXT   | # EXTENDED VERB EXIT                      | 13 |
| 13 |                                                                                                     |        |          | # (THE PURPOS OF UPERROUT +2 EXIT IS      | 14 |
| 14 |                                                                                                     |        |          | # TO PROVIDE AN ERROR EXIT WHICH DOES NOT | 15 |
| 15 |                                                                                                     |        |          | # RESET ANY RESTART GROUPS)               | 16 |
| 16 |                                                                                                     |        |          |                                           | 17 |
| 17 | # 'UPACTOFF' IS A ROUTINE TO TURN OFF UPLINK ACTIVITY LIGHT ON ALL EXITS FROM UPDATE PROGRAM (P27). |        |          |                                           | 18 |
| 18 |                                                                                                     |        |          |                                           | 19 |
| 19 | UPACTOFF                                                                                            | CS     | BIT3     |                                           | 20 |
| 20 |                                                                                                     | EXTEND |          | # TURN OFF UPLINK ACTIVITY LIGHT          | 21 |
| 21 |                                                                                                     | WAND   | DSALMOUT | # (BIT 3 OF CHANNEL 11)                   | 22 |
| 22 |                                                                                                     | TC     | Q        |                                           | 23 |
| 23 |                                                                                                     |        |          |                                           | 24 |
| 24 |                                                                                                     |        |          |                                           | 25 |
| 25 |                                                                                                     |        |          |                                           | 26 |
| 26 |                                                                                                     |        |          |                                           | 27 |
| 27 |                                                                                                     |        |          |                                           | 28 |
| 28 |                                                                                                     |        |          |                                           | 29 |
| 29 |                                                                                                     |        |          |                                           | 30 |
| 30 |                                                                                                     |        |          |                                           | 31 |
| 31 |                                                                                                     |        |          |                                           | 32 |
| 32 |                                                                                                     |        |          |                                           | 33 |
| 33 |                                                                                                     |        |          |                                           | 34 |
| 34 |                                                                                                     |        |          |                                           | 35 |
| 35 |                                                                                                     |        |          |                                           | 36 |
| 36 |                                                                                                     |        |          |                                           | 37 |
| 37 |                                                                                                     |        |          |                                           | 38 |
| 38 |                                                                                                     |        |          |                                           | 39 |
| 39 |                                                                                                     |        |          |                                           | 40 |
| 40 |                                                                                                     |        |          |                                           | 41 |
| 41 |                                                                                                     |        |          |                                           | 42 |
| 42 |                                                                                                     |        |          |                                           | 43 |
| 43 |                                                                                                     |        |          |                                           | 44 |
| 44 |                                                                                                     |        |          |                                           | 45 |
| 45 |                                                                                                     |        |          |                                           | 46 |
| 46 |                                                                                                     |        |          |                                           | 47 |
| 47 |                                                                                                     |        |          |                                           | 48 |
| 48 |                                                                                                     |        |          |                                           | 49 |
| 49 |                                                                                                     |        |          |                                           | 50 |
| 50 |                                                                                                     |        |          |                                           | 51 |
| 51 |                                                                                                     |        |          |                                           | 52 |
| 52 |                                                                                                     |        |          |                                           | 53 |
| 53 |                                                                                                     |        |          |                                           | 54 |
| 54 |                                                                                                     |        |          |                                           | 55 |
| 55 |                                                                                                     |        |          |                                           | 56 |
| 56 |                                                                                                     |        |          |                                           | 57 |
| 57 |                                                                                                     |        |          |                                           | 58 |
| 58 |                                                                                                     |        |          |                                           | 59 |
| 59 |                                                                                                     |        |          |                                           | 60 |
| 60 |                                                                                                     |        |          |                                           | 61 |

BANK 22  
SETLOC RTBCODES  
BANK  
EBANK= XNB  
COUNT\* \$\$/RTB

# LOAD TIME2, TIME1 INTO MPAC:

LOADTIME EXTEND  
DCA TIME2  
TCF SLOAD2

# CONVERT THE SINGLE PRECISION 2'S COMPLEMENT NUMBER ARRIVING IN MPAC (SCALED IN HALF-REVOLUTIONS) TO A  
# DP 1'S COMPLEMENT NUMBER SCALED IN REVOLUTIONS.

CDULOGIC CCS MPAC  
CAF ZERO  
TCF +3  
NOOP  
CS HALF  
TS MPAC +1  
CAF ZERO  
XCH MPAC  
EXTEND  
MP HALF  
DAS MPAC  
TCF DANZIG

# MODE IS ALREADY AT DOUBLE-PRECISION

# FORCE TP SIGN AGREEMENT IN MPAC:

SGNAGREE TC TPAGREE  
TCF DANZIG

# CONVERT THE DP 1'S COMPLEMENT ANGLE SCALED IN REVOLUTIONS TO A SINGLE PRECISION 2'S COMPLEMENT ANGLE  
# SCALED IN HALF-REVOLUTIONS.

1STO2S TC 1TO2SUB  
CAF ZERO  
TS MPAC +1  
TCF NEWMODE

# DO 1STO2S ON A VECTOR OF ANGLES:

V1STO2S TC 1TO2SUB # ANSWER ARRIVES IN A AND MPAC.

DXCH MPAC +5  
DXCH MPAC  
TC 1TO2SUB

```
1
2 TS MPAC +2
3
4 DXCH MPAC +3
5 DXCH MPAC
6 TC 1TO2SUB
7
8 TS MPAC +1
9
10 CA MPAC +5
11 TS MPAC
12
13 TMODE CAF ONE # MODE IS TP.
14 TCF NEWMODE
15
16 # V1STO2S FOR 2 COMPONENT VECTOR. USED BY RR.
17
18 2V1STO2S TC 1TO2SUB
19 DXCH MPAC +3
20 DXCH MPAC
21 TC 1TO2SUB
22 TS L
23 CA MPAC +3
24 TCF SLOAD2
25
26 # SUBROUTINE TO DO DOUBLING AND 1'S TO 2'S CONVERSION:
27
28 1TO2SUB DXCH MPAC # FINAL MPAC +1 UNSPECIFIED.
29 DDOUBL
30 CCS A
31 AD ONE
32 TCF +2
33 COM # THIS WAS REVERSE OF MSU.
34
35 TS MPAC
36 TC Q # AND SKIP ON OVERFLOW.
37
38 INDEX A # OVERFLOW UNCORRECT AND IN MSU.
39 CAF LIMITS
40 ADS MPAC
41 TC Q
42
43 # THE FOLLOWING ROUTINE INCREMENTS IN 2S COMPLEMENT THE REGISTER WHOSE ADDRESS IS IN BUF BY THE 1S COMPL.
44 # QUANTITY FOUND IN TEM2. THIS MAY BE USED TO INCREMENT DESIRED IMU AND OPTICS CDU ANGLES OR ANY OTHER 2S COMPL.
45 # (+0 UNEQUAL TO -0) QUANTITY. MAY BE CALLED BY BANKCALL/SWCALL.
46
47 CDUINC TS TEM2 # 1S COMPL.QUANT. ARRIVES IN ACC. STORE IT
48 INDEX BUF
49 CCS 0 # CHANGE 2S COMPL. ANGLE(IN BUF)INTO 1S
50
51 AD ONE
52 TCF +4
53 AD ONE
```



|    |  |       |        |                                            |    |
|----|--|-------|--------|--------------------------------------------|----|
| 1  |  |       |        |                                            | 1  |
| 2  |  | AD    | ONE    | # OVERFLOW HERE IF 2S COMPL. IS 180 DEG.   | 2  |
| 3  |  | COM   |        |                                            | 3  |
| 4  |  |       |        |                                            | 4  |
| 5  |  | AD    | TEM2   | # SULT MOVES FROM 2ND TO 3D QUAD.(OR BACK) | 5  |
| 6  |  | CCS   | A      | # BACK TO 2S COMPL.                        | 6  |
| 7  |  | AD    | ONE    |                                            | 7  |
| 8  |  | TCF   | +2     |                                            | 8  |
| 9  |  | COM   |        |                                            | 9  |
| 10 |  | TS    | TEM2   | # STORE 14BIT QUANTITY WITH PRESENT SIGN   | 10 |
| 11 |  | TCF   | +4     |                                            | 11 |
| 12 |  | INDEX | A      | # SIGN.                                    | 12 |
| 13 |  | CAF   | LIMITS | # FIX IT,BY ADDING IN 37777 OR 40000       | 13 |
| 14 |  | AD    | TEM2   |                                            | 14 |
| 15 |  |       |        |                                            | 15 |
| 16 |  | INDEX | BUF    |                                            | 16 |
| 17 |  | TS    | 0      | # STORE NEW ANGLE IN 2S COMPLEMENT.        | 17 |
| 18 |  | TC    | Q      |                                            | 18 |
| 19 |  |       |        |                                            | 19 |
| 20 |  |       |        |                                            | 20 |
| 21 |  |       |        |                                            | 21 |
| 22 |  |       |        |                                            | 22 |
| 23 |  |       |        |                                            | 23 |
| 24 |  |       |        |                                            | 24 |
| 25 |  |       |        |                                            | 25 |
| 26 |  |       |        |                                            | 26 |
| 27 |  |       |        |                                            | 27 |
| 28 |  |       |        |                                            | 28 |
| 29 |  |       |        |                                            | 29 |
| 30 |  |       |        |                                            | 30 |
| 31 |  |       |        |                                            | 31 |
| 32 |  |       |        |                                            | 32 |
| 33 |  |       |        |                                            | 33 |
| 34 |  |       |        |                                            | 34 |
| 35 |  |       |        |                                            | 35 |
| 36 |  |       |        |                                            | 36 |
| 37 |  |       |        |                                            | 37 |
| 38 |  |       |        |                                            | 38 |
| 39 |  |       |        |                                            | 39 |
| 40 |  |       |        |                                            | 40 |
| 41 |  |       |        |                                            | 41 |
| 42 |  |       |        |                                            | 42 |
| 43 |  |       |        |                                            | 43 |
| 44 |  |       |        |                                            | 44 |
| 45 |  |       |        |                                            | 45 |
| 46 |  |       |        |                                            | 46 |
| 47 |  |       |        |                                            | 47 |
| 48 |  |       |        |                                            | 48 |
| 49 |  |       |        |                                            | 49 |
| 50 |  |       |        |                                            | 50 |
| 51 |  |       |        |                                            | 51 |
| 52 |  |       |        |                                            | 52 |
| 53 |  |       |        |                                            | 53 |
| 54 |  |       |        |                                            | 54 |
| 55 |  |       |        |                                            | 55 |
| 56 |  |       |        |                                            | 56 |
| 57 |  |       |        |                                            | 57 |
| 58 |  |       |        |                                            | 58 |
| 59 |  |       |        |                                            | 59 |
| 60 |  |       |        |                                            | 60 |





|    |                                                                                          |       |          |                                            |    |
|----|------------------------------------------------------------------------------------------|-------|----------|--------------------------------------------|----|
| 1  | # RTB TO TORQUE GYROS, EXCEPT FOR THE CALL TO IMUSTALL. ECADR OF COMMANDS ARRIVES IN X1. |       |          |                                            | 1  |
| 2  | #                                                                                        |       |          |                                            | 2  |
| 3  |                                                                                          |       |          |                                            | 3  |
| 4  | PULSEIMU                                                                                 | INDEX | FIXLOC   | # ADDRESS OF GYRO COMMANDS SHOULD BE IN X1 | 4  |
| 5  |                                                                                          | CA    | X1       |                                            | 5  |
| 6  |                                                                                          | TC    | BANKCALL |                                            | 6  |
| 7  |                                                                                          | CADR  | IMUPULSE |                                            | 7  |
| 8  |                                                                                          | TCF   | DANZIG   |                                            | 8  |
| 9  |                                                                                          |       |          |                                            | 9  |
| 10 |                                                                                          |       |          |                                            | 10 |
| 11 |                                                                                          |       |          |                                            | 11 |
| 12 |                                                                                          |       |          |                                            | 12 |
| 13 |                                                                                          |       |          |                                            | 13 |
| 14 |                                                                                          |       |          |                                            | 14 |
| 15 |                                                                                          |       |          |                                            | 15 |
| 16 |                                                                                          |       |          |                                            | 16 |
| 17 |                                                                                          |       |          |                                            | 17 |
| 18 |                                                                                          |       |          |                                            | 18 |
| 19 |                                                                                          |       |          |                                            | 19 |
| 20 |                                                                                          |       |          |                                            | 20 |
| 21 |                                                                                          |       |          |                                            | 21 |
| 22 |                                                                                          |       |          |                                            | 22 |
| 23 |                                                                                          |       |          |                                            | 23 |
| 24 |                                                                                          |       |          |                                            | 24 |
| 25 |                                                                                          |       |          |                                            | 25 |
| 26 |                                                                                          |       |          |                                            | 26 |
| 27 |                                                                                          |       |          |                                            | 27 |
| 28 |                                                                                          |       |          |                                            | 28 |
| 29 |                                                                                          |       |          |                                            | 29 |
| 30 |                                                                                          |       |          |                                            | 30 |
| 31 |                                                                                          |       |          |                                            | 31 |
| 32 |                                                                                          |       |          |                                            | 32 |
| 33 |                                                                                          |       |          |                                            | 33 |
| 34 |                                                                                          |       |          |                                            | 34 |
| 35 |                                                                                          |       |          |                                            | 35 |
| 36 |                                                                                          |       |          |                                            | 36 |
| 37 |                                                                                          |       |          |                                            | 37 |
| 38 |                                                                                          |       |          |                                            | 38 |
| 39 |                                                                                          |       |          |                                            | 39 |
| 40 |                                                                                          |       |          |                                            | 40 |
| 41 |                                                                                          |       |          |                                            | 41 |
| 42 |                                                                                          |       |          |                                            | 42 |
| 43 |                                                                                          |       |          |                                            | 43 |
| 44 |                                                                                          |       |          |                                            | 44 |
| 45 |                                                                                          |       |          |                                            | 45 |
| 46 |                                                                                          |       |          |                                            | 46 |
| 47 |                                                                                          |       |          |                                            | 47 |
| 48 |                                                                                          |       |          |                                            | 48 |
| 49 |                                                                                          |       |          |                                            | 49 |
| 50 |                                                                                          |       |          |                                            | 50 |
| 51 |                                                                                          |       |          |                                            | 51 |
| 52 |                                                                                          |       |          |                                            | 52 |
| 53 |                                                                                          |       |          |                                            | 53 |
| 54 |                                                                                          |       |          |                                            | 54 |
| 55 |                                                                                          |       |          |                                            | 55 |
| 56 |                                                                                          |       |          |                                            | 56 |
| 57 |                                                                                          |       |          |                                            | 57 |
| 58 |                                                                                          |       |          |                                            | 58 |
| 59 |                                                                                          |       |          |                                            | 59 |
| 60 |                                                                                          |       |          |                                            | 60 |

# THE SUBROUTINE SIGNMPAC SETS C(MPAC, MPAC +1) TO SIGN(MPAC).  
# FOR THIS, ONLY THE CONTENTS OF MPAC ARE EXAMINED. ALSO +0 YIELDS POSMAX AND -0 YIELDS NEGMAX.

# ENTRY MAY BE BY EITHER OF THE FOLLOWING:  
# 1. LIMIT THE SIZE OF MPAC ON INTERPRETIVE OVERFLOW:

# ENTRY: BOVB

# SIGNMPAC

# 2. GENERATE IN MPAC THE SIGNUM FUNCTION OF MPAC:

# ENTRY: RTB

# SIGNMPAC

# IN EITHER CASE, RETURN IS TO THE NEXT INTERPRETIVE INSTRUCTION IN THE CALLING SEQUENCE.

|          |        |                                            |
|----------|--------|--------------------------------------------|
| SIGNMPAC | EXTEND |                                            |
|          | DCA    | DPOSMAX                                    |
|          | DXCH   | MPAC                                       |
|          | CCS    | A                                          |
| DPMODE   | CAF    | ZERO # SETS MPAC +2 TO ZERO IN THE PROCESS |
|          | TCF    | SLOAD2 +2                                  |
|          | TCF    | +1                                         |
|          | EXTEND |                                            |
|          | DCS    | DPOSMAX                                    |
|          | TCF    | SLOAD2                                     |

# RTB OP CODE NORMUNIT IS LIKE INTERPRETIVE INSTRUCTION UNIT, EXCEPT THAT IT CAN BE DEPENDED ON NOT TO BLOW  
# UP WHEN THE VECTOR BEING UNITIZED IS VERY SMALL -- IT WILL BLOW UP WHEN ALL COMPONENTS ARE ZERO. IF NORMUNIT  
# IS USED AND THE UPPER ORDER HALVES OF ALL COMPONENTS ARE ZERO, THE MAGNITUDE RETURNED IN 36D WILL BE TOO LARGE  
# BY A FACTOR OF 2(13) AND THE SQUARED MAGNITUDE RETURNED AT 34D WILL BE TOO BIG BY A FACTOR OF 2(26).

|          |      |                                                 |
|----------|------|-------------------------------------------------|
| NORMUNX1 | CAF  | ONE                                             |
|          | TCF  | NORMUNIT +1                                     |
| NORMUNIT | CAF  | ZERO                                            |
|          | AD   | FIXLOC                                          |
|          | TS   | MPAC +2                                         |
|          | TC   | BANKCALL # GET SIGN AGREEMENT IN ALL COMPONENTS |
|          | CADR | VECAGREE                                        |
|          | CCS  | MPAC                                            |
|          | TCF  | NOSHIFT                                         |
|          | TCF  | +2                                              |
|          | TCF  | NOSHIFT                                         |
|          | CCS  | MPAC +3                                         |
|          | TCF  | NOSHIFT                                         |
|          | TCF  | +2                                              |
|          | TCF  | NOSHIFT                                         |
|          | CCS  | MPAC +5                                         |
|          | TCF  | NOSHIFT                                         |
|          | TCF  | +2                                              |
|          | TCF  | NOSHIFT                                         |

|    |                               |        |             |                                             |    |
|----|-------------------------------|--------|-------------|---------------------------------------------|----|
| 1  |                               |        |             |                                             | 1  |
| 2  |                               | CA     | MPAC +1     | # SHIFT ALL COMPONENTS LEFT 13              | 2  |
| 3  |                               | EXTEND |             |                                             | 3  |
| 4  |                               | MP     | BIT14       |                                             | 4  |
| 5  |                               | DAS    | MPAC        | # DAS GAINS A LITTLE ACCURACY               | 5  |
| 6  |                               | CA     | MPAC +4     |                                             | 6  |
| 7  |                               | EXTEND |             |                                             | 7  |
| 8  |                               | MP     | BIT14       |                                             | 8  |
| 9  |                               | DAS    | MPAC +3     |                                             | 9  |
| 10 |                               | CA     | MPAC +6     |                                             | 10 |
| 11 |                               | EXTEND |             |                                             | 11 |
| 12 |                               | MP     | BIT14       |                                             | 12 |
| 13 |                               | DAS    | MPAC +5     |                                             | 13 |
| 14 |                               | CAF    | THIRTEEN    |                                             | 14 |
| 15 |                               | INDEX  | MPAC +2     |                                             | 15 |
| 16 |                               | TS     | 37D         |                                             | 16 |
| 17 | OFFTUNIT                      | TC     | POSTJUMP    |                                             | 17 |
| 18 |                               | CADR   | UNIT +1     | # SKIP THE "TC VECAGREE" DONE AT UNIT       | 18 |
| 19 |                               |        |             |                                             | 19 |
| 20 | NOSHIFT                       | CAF    | ZERO        |                                             | 20 |
| 21 |                               | TCF    | OFFTUNIT -2 |                                             | 21 |
| 22 |                               |        |             |                                             | 22 |
| 23 | # RTB VECSGNAG                |        |             | ...FORCES SIGN AGREEMENT OF VECTOR IN MPAC. | 23 |
| 24 |                               |        |             |                                             | 24 |
| 25 | VECSGNAG                      | TC     | BANKCALL    |                                             | 25 |
| 26 |                               | CADR   | VECAGREE    |                                             | 26 |
| 27 |                               | TC     | DANZIG      |                                             | 27 |
| 28 |                               |        |             |                                             | 28 |
| 29 | # *** END OF SKIPPER .087 *** |        |             |                                             | 29 |
| 30 |                               |        |             |                                             | 30 |
| 31 |                               |        |             |                                             | 31 |
| 32 |                               |        |             |                                             | 32 |
| 33 |                               |        |             |                                             | 33 |
| 34 |                               |        |             |                                             | 34 |
| 35 |                               |        |             |                                             | 35 |
| 36 |                               |        |             |                                             | 36 |
| 37 |                               |        |             |                                             | 37 |
| 38 |                               |        |             |                                             | 38 |
| 39 |                               |        |             |                                             | 39 |
| 40 |                               |        |             |                                             | 40 |
| 41 |                               |        |             |                                             | 41 |
| 42 |                               |        |             |                                             | 42 |
| 43 |                               |        |             |                                             | 43 |
| 44 |                               |        |             |                                             | 44 |
| 45 |                               |        |             |                                             | 45 |
| 46 |                               |        |             |                                             | 46 |
| 47 |                               |        |             |                                             | 47 |
| 48 |                               |        |             |                                             | 48 |
| 49 |                               |        |             |                                             | 49 |
| 50 |                               |        |             |                                             | 50 |
| 51 |                               |        |             |                                             | 51 |
| 52 |                               |        |             |                                             | 52 |
| 53 |                               |        |             |                                             | 53 |
| 54 |                               |        |             |                                             | 54 |
| 55 |                               |        |             |                                             | 55 |
| 56 |                               |        |             |                                             | 56 |
| 57 |                               |        |             |                                             | 57 |
| 58 |                               |        |             |                                             | 58 |
| 59 |                               |        |             |                                             | 59 |
| 60 |                               |        |             |                                             | 60 |

```
1 # PROGRAM NAMES: (1) T6JOBCHK MOD. NO. 5 OCTOBER 2, 1967
2 #
3 # MODIFICATION BY: LOWELL G. HULL (A.C.ELECTRONICS)
4 #
5 # THESE PROGRAMS ENABLE THE LM DAP TO CONTROL THE THRUST TIMES OF THE REACTION CONTROL SYSTEM JETS BY USING TIME6.
6 # SINCE THE LM DAP MAINTAINS EXCLUSIVE CONTROL OVER TIME6 AND ITS INTERRUPTS, THE FOLLOWING CONVENTIONS HAVE BEEN
7 # ESTABLISHED AND MUST NOT BE TAMPERED WITH:
8 # 1. NO NUMBER IS EVER PLACED INTO TIME6 EXCEPT BY LM DAP.
9 # 2. NO PROGRAM OTHER THAN LM DAP ENABLES THE TIME6 COUNTER.
10 # 3. TO USE TIME6, THE FOLLOWING SEQUENCE IS ALWAYS EMPLOYED:
11 # A. A POSITIVE (NON-ZERO) NUMBER IS STORED IN TIME6.
12 # B. THE TIME6 CLOCK IS ENABLED.
13 # C. TIME6 IS INTERROGATED AND IS:
14 # I. NEVER FOUND NEGATIVE (NON-ZERO) OR +0.
15 # II. SOMETIMES FOUND POSITIVE (BETWEEN 1 AND 240D) INDICATING THAT IT IS ACTIVE.
16 # III. SOMETIMES FOUND POSMAX INDICATING THAT IT IS INACTIVE AND NOT ENABLED.
17 # IV. SOMETIMES FOUND NEGATIVE ZERO INDICATING THAT:
18 # A. A T6RUPT IS ABOUT TO OCCUR AT THE NEXT DINC, OR
19 # B. A T6RUPT IS WAITING IN THE PRIORITY CHAIN, OR
20 # C. A T6RUPT IS IN PROCESS NOW.
21 #
22 # 4. ALL PROGRAMS WHICH OPERATE IN EITHER INTERRUPT MODE OR WITH INTERRUPT INHIBITED MUST CALL T6JOBCHK
23 # EVERY 5 MILLISECONDS TO PROCESS A POSSIBLE WAITING T6RUPT BEFORE IT CAN BE HONORED BY THE HARDWARE.
24 # (5. PROGRAM JTLST, IN Q,R-AXES, HANDLES THE INPUT LIST.)
25 #
26 # T6JOBCHK CALLING SEQUENCE:
27 # L TC T6JOBCHK
28 # L+1 (RETURN)
29 #
30 # DOT6RUPT CALLING SEQUENCE:
31 # DXCH ARUPT # T6RUPT LEAD IN AT LOCATION 4004.
32 # EXTEND
33 # DCA T6ADR
34 # DTCB
35 #
36 # SUBROUTINES CALLED: DOT6RUPT CALLS T6JOBCHK.
37 #
38 # NORMAL EXIT MODES: T6JOBCHK RETURNS TO L +1.
39 # DOT6RUPT TRANSFERS CONTROL TO RESUME.
40 #
41 # ALARM/ABORT MODES: NONE.
42 #
43 # INPUT: TIME6 NXT6ADR OUTPUT: TIME6 NXT6ADR CHANNEL 5
44 # T6NEXT T6NEXT +1 T6NEXT T6NEXT +1 CHANNEL 6
45 # T6FURTHA T6FURTHA +1 T6FURTHA T6FURTHA +1 BIT15/CH13
46 #
47 # DEBRIS: T6JOBCHK CLOBBERS A. DOT6RUPT CLOBBERS NOTHING.
48
49 BLOCK 02
50
51
52
53
54
55
56
57
58
59
60
```

```
1
2 BANK 17
3 SETLOC DAPS2
4
5 BANK
6 EBANK= T6NEXT
7 COUNT* $$/DAPT6
8
9 T6JOBCHK CCS TIME6 # CHECK TIME6 FOR WAITING T6RUPT:
10 TC Q # NONE: CLOCK COUNTING DOWN.
11
12 TC CCSHOLE
13 TC T6JOBCHK +3
14
15 # CONTROL PASSES TO T6JOB ONLY WHEN C(TIME6) = -0 (I.E., WHEN A T6RUPT MUST BE PROCESSED).
16
17 T6JOB CAF POSMAX # DISABLE CLOCK: NEEDED SINCE RUPT OCCURS
18 EXTEND # 1 DINC AFTER T6 = 77777. FOR 625 MUSECS
19 WAND CHAN13 # MUST NOT HAVE T6 = +0 WITH ENABLE SET
20
21 CA POSMAX
22 ZL
23 DXCH T6FURTHA
24 DXCH T6NEXT
25 LXCH NXT6ADR
26 TS TIME6
27
28 AD PRI037
29 TS A
30
31 TCF ENABLET6
32 CA POSMAX
33 TS TIME6
34
35 ENABLET6 TCF GOCH56
36 CA BIT15
37 EXTEND
38
39 WOR CHAN13
40 CA T6NEXT
41 AD PRI037
42
43 TS A
44
45 TCF GOCH56
46 CA POSMAX
47
48 GOCH56 TS T6NEXT
49 INDEX L
50 TCF WRITEP -1
51
52 BLOCK 02
53 SETLOC FFTAG9
54
55 BANK
56 EBANK= CDUXD
57 COUNT* $$/DAPT6
58
59 WRITEP CA NEXTP
60 EXTEND
61 WRITE CHAN6
```



```
1
2 BANK 20
3 SETLOC DAPS3
4 BANK
5
6 EBANK= CDUXD
7 COUNT* $$/DAPIF
8
9 # MOD 0 DATE 11/15/66 BY GEORGE W. CHERRY
10 # MOD 1 DATE 1/23/67 MODIFICATION BY PETER ADLER
11 #
12 # FUNCTIONAL DESCRIPTION
13 # HEREIN ARE A COLLECTION OF SUBROUTINES WHICH ALLOW MISSION CONTROL PROGRAMS TO CONTROL THE MODE
14 # AND INTERFACE WITH THE DAP.
15 #
16 # CALLING SEQUENCES
17 # IN INTERRUPT OR WITH INTERRUPT INHIBITED
18 # TC IBNKCALL
19 # FCADR ROUTINE
20 # IN A JOB WITHOUT INTERRUPT INHIBITED
21 # INHINT
22 # TC IBNKCALL
23 # FCADR ROUTINE
24 # RELINT
25 #
26 # OUTPUT
27 # SEE INDIVIDUAL ROUTINES BELOW
28 #
29 # DEBRIS
30 # A, L, AND SOMETIMES MDUETEMP ODE NOT IN PULSES MODE
```

## # SUBROUTINE NAMES:

# SETMAXDB, SETMINDB, RESTORDB, PFLITEDB

# MODIFIED: 30 JANUARY 1968 BY P S WEISSMAN TO CREATE RESTORDB.

# MODIFIED: 1 MARCH 1968 BY P S WEISSMAN TO SAVE EBANK AND CREATE PFLITEDB

#

## # FUNCTIONAL DESCRIPTION:

# SETMAXDB -- SET DEADBAND TO 5.0 DEGREES

# SETMINDB -- SET DEADBAND TO 0.3 DEGREE

# RESTORDB -- SET DEADBAND TO MAX OR MIN ACCORDING TO SETTINGS OF DBSELECT BIT OF DAPBOOLS

# PFLITEDB -- SET DEADBAND TO 1.0 DEGREE AND ZERO THE COMMANDED ATTITUDE CHANGE AND COMMANDED RATE

#

# ALL ENTRIES SET UP A NOVAC JOB TO DO 1/ACCS SO THAT THE TJETLAW SWITCH CURVES ARE POSITIONED TO REFLECT THE NEW DEADBAND. IT SHOULD BE NOTED THAT THE DEADBAND REFERS TO THE ATTITUDE IN THE P-, U-, AND V-AXES.

#

# SUBROUTINE CALLED: NOVAC

#

# CALLING SEQUENCE: SAME AS ABOVE

# OR TC RESTORDB +1 FROM ALLCOAST

#

# DEBRIS: A, L, Q, RUPTREG1, (ITEMPS IN NOVAC)

RESTORDB CAE DAPBOOLS # DETERMINE CREW-SELECTED DEADBAND.

MASK DBSELECT

EXTEND

BZF SETMINDB

SETMAXDB CAF WIDEDB # SET 5 DEGREE DEADBAND.

+1

TS

DB

CALLACCS EXTEND # SET UP JOB TO RE-POSITION SWITCH CURVES.

QXCH RUPTREG1

CAF PRI027

TC NOVAC

EBANK= AOSQ

2CADR 1/ACCJOB

TC RUPTREG1 # RETURN TO CALLER.

SETMINDB CAF NARROWDB # SET 0.3 DEGREE DEADBAND.

TCF SETMAXDB +1

PFLITEDB EXTEND # THE RETURN FROM CALLACCS IS TO RUPTREG1.

QXCH RUPTREG1

TC ZATTEROR

CAF POWERDB

TS DB

TCF CALLACCS

NARROWDB OCTAL 00155 # 0.3 DEGREE SCALED AT 45.



WIDEDB            OCTAL    03434            # 5.0 DEGREES SCALED AT 45.  
POWERDB          DEC      .02222            # 1.0 DEGREE SCALED AT 45.

ZATTEROR          CAF      EBANK6  
                     XCH      EBANK

                     TS       L                # SAVE CALLERS EBANK IN L.  
                     CAE      CDUX  
                     TS       CDUXD

                     CAE      CDUY  
                     TS       CDUYD  
                     CAE      CDUZ

                     TS       CDUZD  
                     TCF      STOPRATE +3

STOPRATE          CAF      EBANK6  
                     XCH      EBANK

                     TS       L                # SAVE CALLERS EBANK IN L.

+3                  CAF      ZERO  
                     TS       OMEGAPD  
                     TS       OMEGAQD

                     TS       OMEGARD  
                     TS       DELCDUX  
                     TS       DELCDUY

                     TS       DELCDUZ  
                     TS       DELPEROR  
                     TS       DELQEROR

                     TS       DELREROR  
                     LXCH     EBANK            # RESTORE CALLERS EBANK.  
                     TC       Q

# SUBROUTINE NAME:        ALLCOAST  
# WILL BE CALLED BY FRESH STARTS AND ENGINE OFF ROUTINES.

#  
# CALLING SEQUENCE:        (SAME AS ABOVE)  
#

# EXIT:                    RETURN TO Q.

# SUBROUTINES CALLED:     STOPRATE, RESTORDB, NOVAC

#  
# ZERO:                    (FOR ALL AXES) AOS, ALPHA, AOSTERM, OMEGAD, DELCDU, DELEROR  
#

# OUTPUT:                  DRIFTBIT/DAPBOOLS, OE, JOB TO DO 1/ACCS

#  
# DEBRIS:                  A, L, Q, RUPTREG1, RUPTREG2, (ITEMPS IN NOVAC)

ALLCOAST            EXTEND            # SAVE Q FOR RETURN  
                     QXCH      RUPTREG2



|    |  |      |             |                                            |    |
|----|--|------|-------------|--------------------------------------------|----|
| 1  |  |      |             |                                            | 1  |
| 2  |  | TC   | STOPRATE    | # CLEAR RATE INTERFACE. RETURN WITH A=0    | 2  |
| 3  |  | LXCH | EBANK       | # AND L=EBANK6. SAVE CALLER'S EBANK.       | 3  |
| 4  |  | TS   | AOSQ        |                                            | 4  |
| 5  |  | TS   | AOSQ +1     |                                            | 5  |
| 6  |  | TS   | AOSR        |                                            | 6  |
| 7  |  | TS   | AOSR +1     |                                            | 7  |
| 8  |  | TS   | ALPHAQ      | # FOR DOWNLIST.                            | 8  |
| 9  |  | TS   | ALPHAR      |                                            | 9  |
| 10 |  | TS   | AOSQTERM    |                                            | 10 |
| 11 |  | TS   | AOSRTERM    |                                            | 11 |
| 12 |  | LXCH | EBANK       | # RESTORE EBANK (EBANK6 NO LONGER NEEDED)  | 12 |
| 13 |  |      |             |                                            | 13 |
| 14 |  | CS   | DAPBOOLS    | # SET UP DRIFTBIT                          | 14 |
| 15 |  | MASK | DRIFTBIT    |                                            | 15 |
| 16 |  | ADS  | DAPBOOLS    |                                            | 16 |
| 17 |  | TC   | RESTORDB +1 | # RESTORE DEADBANK TO CREW-SELECTED VALUE. | 17 |
| 18 |  |      |             |                                            | 18 |
| 19 |  | TC   | RUPTREG2    | # RETURN.                                  | 19 |
| 20 |  |      |             |                                            | 20 |
| 21 |  |      |             |                                            | 21 |
| 22 |  |      |             |                                            | 22 |
| 23 |  |      |             |                                            | 23 |
| 24 |  |      |             |                                            | 24 |
| 25 |  |      |             |                                            | 25 |
| 26 |  |      |             |                                            | 26 |
| 27 |  |      |             |                                            | 27 |
| 28 |  |      |             |                                            | 28 |
| 29 |  |      |             |                                            | 29 |
| 30 |  |      |             |                                            | 30 |
| 31 |  |      |             |                                            | 31 |
| 32 |  |      |             |                                            | 32 |
| 33 |  |      |             |                                            | 33 |
| 34 |  |      |             |                                            | 34 |
| 35 |  |      |             |                                            | 35 |
| 36 |  |      |             |                                            | 36 |
| 37 |  |      |             |                                            | 37 |
| 38 |  |      |             |                                            | 38 |
| 39 |  |      |             |                                            | 39 |
| 40 |  |      |             |                                            | 40 |
| 41 |  |      |             |                                            | 41 |
| 42 |  |      |             |                                            | 42 |
| 43 |  |      |             |                                            | 43 |
| 44 |  |      |             |                                            | 44 |
| 45 |  |      |             |                                            | 45 |
| 46 |  |      |             |                                            | 46 |
| 47 |  |      |             |                                            | 47 |
| 48 |  |      |             |                                            | 48 |
| 49 |  |      |             |                                            | 49 |
| 50 |  |      |             |                                            | 50 |
| 51 |  |      |             |                                            | 51 |
| 52 |  |      |             |                                            | 52 |
| 53 |  |      |             |                                            | 53 |
| 54 |  |      |             |                                            | 54 |
| 55 |  |      |             |                                            | 55 |
| 56 |  |      |             |                                            | 56 |
| 57 |  |      |             |                                            | 57 |
| 58 |  |      |             |                                            | 58 |
| 59 |  |      |             |                                            | 59 |
| 60 |  |      |             |                                            | 60 |

# THE DAPIDLER PROGRAM IS STARTED BY FRESH START AND RESTART. THE DAPIDLER PROGRAM IS DONE 10 TIMES  
# PER SECOND UNTIL THE ASTRONAUT DESIRES THE DAP TO WAKE UP, AND THE IMU AND CDUS ARE READY FOR USE BY THE DAP.  
# THE NECESSARY INITIALIZATION OF THE DAP IS DONE BY THE DAPIDLER PROGRAM.

BANK 16  
SETLOC DAPS1

BANK

EBANK= AOSQ

COUNT\* \$\$/DAPID

CHEKBITS

EXTEND

READ CHAN31

# IF BOTH BIT13 AND BIT14 ARE ONE, THEN

COM

# THE MODE SELECT SWITCH IS IN THE OFF

MASK

BIT13-14

# POSITION, AND SO THE DAP SHOULD BE OFF,

EXTEND

# WITH NO ATTITUDE ERROR DISPLAY.

BZF

MOREIDLE

CS

IMODES33

MASK

BIT6

CCS

A

TCF

JUMPDSP

CS

RCSFLAGS

# IMU NOT USABLE. SET UP INITIALIZATION

MASK

BIT3

# FLAG FOR ATT ERROR DISPLAY ROUTINE.

ADS

RCSFLAGS

TCF

SHUTDOWN

CHEKMORE

CAF

BIT10

# BIT 10 OF 30 IS PGNCS CONTROL OF S/C

EXTEND

RAND

CHAN30

# BITS IN 30 ARE INVERTED

CCS

A

TCF

MOREIDLE

RETURN

## # DAPIDLER ENTRY.

|          |                        |                               |                                                                                                             |
|----------|------------------------|-------------------------------|-------------------------------------------------------------------------------------------------------------|
| DAPIDLER | LXCH<br>EXTEND<br>QXCH | BANKRUPT<br>QRUPT             | # INTERRUPT LEAD INS (CONTINUED)                                                                            |
|          | CA<br>MASK             | RCSFLAGS<br>BIT13             |                                                                                                             |
|          | CCS<br>TCF<br>CA       | A<br>CHECKUP<br>BIT13         | # CHECK IF 1/ACCJOB HAS BEEN SET UP SINCE<br># THE LAST FRESH START OR RESTART.                             |
|          | ADS<br>CAF<br>TC       | RCSFLAGS<br>PRIO27<br>NOVAC   | # BIT 13 IS 1.<br># SET UP JOB TO DO A LITTLE INITIALIZATION<br># AND EXECUTE 1/ACCS.                       |
|          | EBANK=<br>2CADR        | AOSQ<br>1/ACCSET              | # (WILL BRANCH TO MOREIDLE ON ACCSOKAY)                                                                     |
| CHECKUP  | TC                     | CHEKBITS                      | # CHECK TO SEE IF LM DAP IS TO GO ON AND<br># DO ERROR DISPLAY.                                             |
|          | CAE<br>MASK<br>EXTEND  | DAPBOOLS<br>ACCSOKAY          | # IF 1/ACCS HAS NOT BEEN COMPLETED, IDLE.<br># NOTE: ONLY FRESH START AND RESTART<br># KNOCK THIS BIT DOWN. |
|          | BZF                    | MOREIDLE                      |                                                                                                             |
| STARTDAP | TC                     | IBNKCALL                      | # ZERO ATTITUDE ERROR AND DESIRED RATES.                                                                    |
|          | FCADR<br>CAF<br>TS     | ZATTEROR<br>ZERO<br>TJP       | # ***** INITIALIZE: *****                                                                                   |
|          | TS<br>TS<br>TS         | TJU<br>TJV<br>OMEGAP          | # RATES IN BODY (PILOT) COORDINATES.                                                                        |
|          | TS<br>TS<br>TS         | OMEGAQ<br>OMEGAR<br>TRAPEDP   |                                                                                                             |
|          | TS<br>TS<br>TS         | TRAPEDQ<br>TRAPEDR<br>AOSQ    | # OFFSET ACCELERATION ESTIMATES.                                                                            |
|          | TS<br>TS<br>TS         | AOSQ +1<br>AOSR<br>AOSR +1    |                                                                                                             |
|          | TS<br>TS<br>TS         | ALPHAQ<br>ALPHAR<br>NEGUQ     | # COPIES OF OFFSET ESTIMATES FOR DOWNLIST.                                                                  |
|          | TS<br>TS<br>TS         | NEGUR<br>AOSQTERM<br>AOSRTERM | # QRAXIS RATE DERIVATION TERMS AND KALMAN<br># FILTER INITIALIZATION TERMS.                                 |
|          | TS<br>TS               | QACCDOT<br>RACCDOT            | # DESCENT ACCELERATION DERIVATIVE EST.                                                                      |

```
1
2 TS ALLOWGTS # AOSTASK FLAG FOR QRAXIS RCS CONTROL USE.
3 TS COTROLER # DO TRYGTS ON FIRST PASS (WILL GO TO RCS)
4 TS INGTS # RECOGNIZE FIRST GTS PASS AS SUCH.
5 TS QGIMTIMR # STOP GIMBAL DRIVES. (PROBABLY WOULD BE
6 TS RGIMTIMR # GOOD ENOUGH JUST TO INACTIVATE TIMERS)
7 TS OLDPMIN # MINIMUM IMPULSE MODE ERASABLES
8 TS OLDQRMIN
9 TS PJETCTR # INITIALIZE DOCKED JET INHIBITION
10 TS UJETCTR # COUNTERS
11 TS VJETCTR
12 CALLGMBL EQUALS BIT5 # RCSFLAGS INITIALIZATION.
13 CS MANFLAG
14 MASK RCSFLAGS # NEGUQ(R) HAVE BEEN GENERATED.
15 TS RCSFLAGS
```

# SET UP "OLD" MEASURED CDU ANGLES:

```
19 EXTEND
20 DCA CDUX # OLDXFORP AND OLDYFORP
21 DXCH OLDXFORP
22 CA CDUZ
23 TS OLDZFORQ
24 CS RCSFLAGS
25 MASK BIT12
26 ADS RCSFLAGS # BIT 12 SET TO 1.
27 CA FOUR
28 TS SKIPU
29 TS SKIPV
30 CA POSMAX
31 TS TIME6
32 TS T6NEXT
33 TS T6FURTHA
34 CA ZERO
35 TS T6NEXT +1
36 TS T6FURTHA +1
37 TS NXT6ADR
38 TS NEXTP
39 TS NEXTU
40 TS NEXTV
41 CS TEN
42 TS DAPZRPT # JASK NOT IN PROGRESS, INITIALIZE NEG.
43 CA TWO
44 TS NPTRAPS
45 TS NQTRAPS
46 TS NRTRAPS
47 EXTEND
48 DCA PAXADIDL
49 SETTIME5 DXCH T5ADR
50 CAF MS100
51 TS TIME5
```

```
1
2 TCF RESUME
3 EBANK= AOSQ
4 IDLERADR 2CADR DAPIDLER
5
6 MOREIDLE TC IBNKCALL # CALCULATE Q,R-AXES ATTITUDE ERRORS.
7 CADR QERRCALC
8
9 TC IBNKCALL
10 CADR CALCPERR # CALCULATE P AXIS ATTITUDE ERRORS.
11
12 SHUTDOWN EXTEND
13 DCA IDLERADR
14 DXCH T5ADR
15
16 CAF ZERO # KILL ANY POSSIBLE JET REQUESTS
17 TS NEXTP
18 TS NEXTU
19 TS NEXTV
20 EXTEND
21 WRITE CHAN5 # COMMAND JETS OFF.
22
23 EXTEND
24 WRITE CHAN6
25 CS BGIM23 # TURN TRIM GIMBAL OFF
26
27 EXTEND
28 WAND CHAN12
29 TCF SETTIME5 # RETURN IN 100 MSEC.
30
31 MANFLAG OCT 03021
32 BGIM23 OCTAL 07400
33
34 PAXADIDL EBANK= OMEGAP
35 2CADR PAXIS
36
37 MS100 = OCT37766
38 COSMG = ITEMP1
39 JUMPDSP EXTEND # TRANSFER TO BANK 20
40
41 DCA DSPCADR # FOR ATTITUDE ERROR DISPLAYS
42 DTCB
43
44 DSPCADR EBANK= AK
45 2CADR ALTDSPLY
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1
2 BANK 20
3 SETLOC DAPS3
4
5 BANK
6 COUNT* $$/NEEDL
7
8 # PROGRAM: ALTDSPY
9 #
10 # MOD 0. 6 DEC 1967
11
12 #
13 # AUTHOR: CRAIG WORK, DON KEENE, MIT IL
14 #
15 # MOD 3 BY DON KEENE AUG 1, 1968 MOVED PROGRAM TO BANK 20
16 #
17 # PROGRAM DESCRIPTION:
18 #
19 # ALTDSPY REVERSES THE DSPYALT BIT OF RCSFLAGS EACH TIME IT IS CALLED, WHICH IS PRESUMABLY EVERY 100 MS.
20 # IF THE REVERSED BIT IS ONE, NEEDLER IS CALLED TO DISPLAY ATTITUDE ERRORS. IF THE BIT IS ZERO, THE ATTITUDE ERR-
21 # ORS ARE CALCULATED AS 1) DAP FOLLOWING ERRORS, IF NEEDLFLG = 0, AND 2) TOTAL ATTITUDE ERRORS FOR NEEDLFLG = 1.
22 #
23 # WARNING: ALTDSPY MAY ONLY BE CALLED WITH INTERRUPT INHIBITED.
24 #
25 # WARNING: EBANK MUST BE SET TO 6 WHEN USING THIS ROUTINE.
26 #
27 # INPUT: RCSFLAGS AND 1) IF NEEDLFLG=0, INPUT PERROR,QERROR,ERROR.
28 # 2) IF NEEDLFLG=1, INPUT CPHI,CTHETA,CPSI,CDUX,CDUY,CDUZ,M11,M21,M32,M22,M32. (GPMATRIX)
29 #
30 # OUTPUTS: RCSFLAGS WITH DSPYALT REVERSED,AK,AK1,AK2,+ NEEDLER OUTPUTS.
31 #
32 # ENTRY: TCF ALTDSPY
33 #
34 # EXIT: TCF CHEKMORE
35 #
36 # ALARM OR ABORT EXITS: NONE
37 #
38 # SUBPROGRAMS CALLED: NEEDLER, OVERSUB2
39 #
40 # DEBRIS: A,L,AND NEEDLER DEBRIS.
41
42 ALTDSPY CA RCSFLAGS # INVERT THE DISPLAY ALTERNATION BIT.
43 TS L
44 CA DSPYALT
45 EXTEND
46 RXOR LCHAN
47 TS RCSFLAGS
48
49 MASK DSPYALT
50 CCS A # IS ALTERNATION FLAG ZERO?
51 TCF NEEDLER
52
53 CAE FLAGWRDO # NEEDLFLG WILL INDICATE TOTAL OR DAP AT-
```

```
1 MASK NEEDLBIT # TITUDE ERROR DISPLAY REQUEST.
2 CCS A
3
4 TCF DSPLYTOT # TOTAL ERROR IS NEEDED IN AK,AK +1,AK +2
5
6 CS QERROR # YES. DISPLAY ATT ERRORS ON THE ,-BALL.
7 TS AK +1 # ERROR COMPLEMENTS ARE INPUT TO NEEDLER.
8 CS RERROR
9 TS AK +2
10 CS PERROR
11 XCH AK
12
13 TCF RETNMORE # DISPLAY THESE THE NEXT TIME THROUGH
14
15 # CALCULATE GIMBAL ANGLE TOTAL ERRORS, RESOLVE INTO PILOT AXES, STORE TOTAL ERRORS FOR NEEDLER. Q-AXIS FIRST.
16
17 DSPLYTOT EXTEND
18 QXCH ITEMP1 # SAVE Q FOR CHEKBITS RETURN.
19
20 CA CTHETA # DESIRED ATTITUDE, Y-AXIS, 2'S COMP.
21 EXTEND # SUBTRACT CURRENT ATTITUDE.
22 MSU CDUY # DIFFERENCE SCALED AT PI, 1'S COMP.
23 TS AK # SAVE FOR R-ERROR CALCULATION.
24 EXTEND
25 MP M21 # (CTHETA-CDUY)*M21 SCALED AT PI RADIANS.
26 XCH AK +1 # STORE FIRST TERM OF Q ERROR.
27 CA CPSI # DESIRED ATTITUDE,Z-AXIS, 2'S COMP.
28 EXTEND # SUBTRACT CURRENT ATTITUDE.
29 MSU CDUZ # DIFFERENCE SCALED AT PI, 1'S COMP.
30 TS AK +2 # SAVE Z-AXIS TERM FOR R ERROR CALCULATION
31 EXTEND
32 MP M22 # (CPSI-CDUZ)*M22, SCALED AT PI RADIANS.
33 AD AK +1 # Q ERROR COMPLETE , AT PI RAD.
34 TC OVERSUB2 # PIN NEEDLES IN CASE OF OVERFLOW
35 TS AK +1
36
37 # R ERROR CALCULATION NEXT.
38
39 CA AK # Y-AXIS DIFFERENCE STORED BY Q-AXIS CALC.
40 EXTEND
41 MP M31 # (CTHETA-CDUY)*M31, SCALED AT PI RADIANS.
42 XCH AK +2 # FIRST TERM OF R ERROR.
43 # Z-AXIS DIFFERENCE, STORED BY A CALC. IS
44 EXTEND # RECOVERED BY THE EXCHANGE.
45 MP M32 # (CPSI-CDUZ)*M32, SCALED AT PI RADIANS.
46 AD AK +2 # R ERROR COMPLETE , AT PI RAD.
47 TC OVERSUB2 # PIN NEEDLES IN CASE OF OVERFLOW.
48 TS AK +2
49
50 # NOW CALCULATE P ERROR. (NOTE THAT M13 = 1, SCALED AT 1, SO THE MULTIPLICATION IS BY-PASSED.)
```



|    |                                          |          |                                            |  |
|----|------------------------------------------|----------|--------------------------------------------|--|
| 1  |                                          |          |                                            |  |
| 2  | CA                                       | AK       | # Y-AXIS DIFFERENCE STORED BY Q AXIS CALC. |  |
| 3  | EXTEND                                   |          |                                            |  |
| 4  | MP                                       | M11      | # (CTHETA-CDUY)*M11 SCALED AT PI RADIANS.  |  |
| 5  | XCH                                      | AK       | # FIRST TERM OF P ERROR IN AK, AT PI RAD.  |  |
| 6  | CAE                                      | CPHI     | # DESIRED ATTITUDE, X-AXIS, 2'S COMP.      |  |
| 7  | EXTEND                                   |          | # SUBTRACT CURRENT X ATTITUDE.             |  |
| 8  | MSU                                      | CDUX     | # X-AXIS DIFFERENCE, 1'S COMP, AT PI RAD.  |  |
| 9  |                                          |          |                                            |  |
| 10 | # M13 = 1, SO BYPASS THE MULTIPLICATION. |          |                                            |  |
| 11 | #                                        | EXTEND   |                                            |  |
| 12 | #                                        | MP       | M13 (CPHI-CDUX)*M13 SCALED AT PI RADIANS.  |  |
| 13 |                                          |          |                                            |  |
| 14 | AD                                       | AK       | # P ERROR COMPLETE , SCALED AT PI RAD      |  |
| 15 | TC                                       | OVERSUB2 | # PIN NEEDLES IN CASE OF OVERFLOW.         |  |
| 16 | TS                                       | AK       |                                            |  |
| 17 |                                          |          |                                            |  |
| 18 | EXTEND                                   |          |                                            |  |
| 19 | QXCH                                     | ITEMP1   | # RESTORE Q FOR CHEKBITS RETURN.           |  |
| 20 |                                          |          |                                            |  |
| 21 | TCF                                      | RETNMORE | # DISPLAY THESE THE NEXT TIME THROUGH      |  |
| 22 |                                          |          |                                            |  |
| 23 |                                          |          |                                            |  |
| 24 |                                          |          |                                            |  |
| 25 |                                          |          |                                            |  |
| 26 |                                          |          |                                            |  |
| 27 |                                          |          |                                            |  |
| 28 |                                          |          |                                            |  |
| 29 |                                          |          |                                            |  |
| 30 |                                          |          |                                            |  |
| 31 |                                          |          |                                            |  |
| 32 |                                          |          |                                            |  |
| 33 |                                          |          |                                            |  |
| 34 |                                          |          |                                            |  |
| 35 |                                          |          |                                            |  |
| 36 |                                          |          |                                            |  |
| 37 |                                          |          |                                            |  |
| 38 |                                          |          |                                            |  |
| 39 |                                          |          |                                            |  |
| 40 |                                          |          |                                            |  |
| 41 |                                          |          |                                            |  |
| 42 |                                          |          |                                            |  |
| 43 |                                          |          |                                            |  |
| 44 |                                          |          |                                            |  |
| 45 |                                          |          |                                            |  |
| 46 |                                          |          |                                            |  |
| 47 |                                          |          |                                            |  |
| 48 |                                          |          |                                            |  |
| 49 |                                          |          |                                            |  |
| 50 |                                          |          |                                            |  |
| 51 |                                          |          |                                            |  |
| 52 |                                          |          |                                            |  |
| 53 |                                          |          |                                            |  |
| 54 |                                          |          |                                            |  |
| 55 |                                          |          |                                            |  |
| 56 |                                          |          |                                            |  |
| 57 |                                          |          |                                            |  |
| 58 |                                          |          |                                            |  |
| 59 |                                          |          |                                            |  |
| 60 |                                          |          |                                            |  |

# FDAI ATTITUDE ERROR DISPLAY SUBROUTINE

# PROGRAM DESCRIPTION: D. KEENE 5/24/67

# MOD 1 BY CRAIG WORK, 12 DEC 67

# MOD 2 BY CRAIG WORK, 6 APRIL 68 CONVERTS ATTITUDE ERROR DISPLAY SCALING FROM 16 7/8 DEG. TO 42 3/16 DEGREES.

# THIS SUBROUTINE IS USED TO DISPLAY ATTITUDE ERRORS ON THE FDAI VIA THE DIGITAL TO ANALOG CONVERTERS (DACS)  
# IN THE CDUS. CARE IS TAKEN TO METER OUT THE APPROPRIATE NUMBER OF PULSES TO THE IMU ERROR COUNTERS AND PREVENT  
# OVERFLOW, TO CONTROL THE RELAY SEQUENCING, AND TO AVOID INTERFERENCE WITH THE COARSE ALIGN LOOP WHICH ALSO USES  
# THE DACS.

# CALLING SEQUENCE:

# DURING THE INITIALIZATION SECTION OF THE USER'S PROGRAM, BIT3 OF RCSFLAGS SHOULD BE SET TO INITIATE THE  
# TURN-ON SEQUENCE WITHIN THE NEEDLES PROGRAM:

# CS RCSFLAGS IN EBANK6

# MASK BIT3

# ADS RCSFLAGS

# THEREAFTER, THE ATTITUDE ERRORS GENERATED BY THE USER SHOULD BE TRANSFERRED TO THE FOLLOWING LOCATIONS IN EBANK6:

# AK SCALED 180 DEGREES NOTE: THESE LOCATIONS ARE SUBJECT

# AK1 SCALED 180 DEGREES TO CHANGE

# AK2 SCALED 180 DEGREES

# FULL SCALED DEFLECTION OF THE NEEDLES CORRESPONDS TO 5 1/16 DEGREES, WHILE 384 BITS IN THE IMU ERROR COUNTER  
# CORRESPONDS TO 42 3/16 DEGREES. (DAC MAXIMUM CAPACITY IS 384 BITS.) 46 BITS EFFECTIVELY PIN THE NEEDLES.

# A CALL TO NEEDLER WILL THEN UPDATE THE DISPLAY:

# INHINT

# TC IBNKCALL NOTE: EBANK SHOULD BE SET TO E6

# CADR NEEDLER

# RELINT

# THIS PROCESS SHOULD BE REPEATED EACH TIME THE ERRORS ARE UPDATED. AT LEAST 3 PASSES THRU THE PROGRAM ARE  
# REQUIRED BEFORE ANYTHING IS ACTUALLY DISPLAYED ON THE ERROR METERS.

# NOTE: EACH CALL TO NEEDLER MUST BE SEPARATED BY AT LEAST 50MS TO ASSURE PROPER RELAY SEQUENCING.

# ERASABLES USED:

# AK CDUXCMD

# AK1 CDUYCMD

# AK2 CDUZCMD

# EDRIVEX A,L,Q

# EDRIVEY T5TEMP

# EDRIVEZ DINDX

```

#
SWITCHES: RCSFLAGS BITS 3,2
#
I/O CHANNELS: CHAN12 BIT 4 (COARSE ALIGN - READ ONLY)
CHAN12 BIT 6 (IMU ERROR COUNTER ENABLE)
CHAN14 BIT 13,14,15 (DAC ACTIVITY)
#
SIGN CONVENTION< AK = THETAC - THETA
WHERE THETAC = COMMAND ANGLE
THETA = PRESENT ANGLE

NEEDLER CA RCSFLAGS
 MASK SIX
 EXTEND
 BZF NEEDLES3
 MASK BIT3
 EXTEND
 BZF NEEDLER2 # BIT3 = 0, BIT2 = 1

 CS BIT6 # FIRST PASS BIT3 = 1
 EXTEND # DISABLE IMU ERROR COUNTER TO ZERO DACS
NEEDLE11 WAND CHAN12 # MUST WAIT AT LEAST 60 MS BEFORE
 CS ZERO # ENABLING COUNTERS.
 TS AK # ZERO THE INPUTS ON FIRST PASS
 TS AK1
 TS AK2
 TS EDRIVEX # ZERO THE DISPLAY REGISTERS
 TS EDRIVEY
 TS EDRIVEZ
 TS CDUXCMD # ZERO THE OUT COUNTERS
 TS CDUYCMD
 TS CDUZCMD
 CS SIX # RESET RCSFLAGS FOR PASS2
 MASK RCSFLAGS
 AD BIT2
 TS RCSFLAGS
 TCF RETNMORE

NEEDLER2 CAF BIT6 # ENABLE IMU ERROR COUNTERS
 EXTEND
 WOR CHAN12
 CS SIX # RESET RCSFLAGS TO DISPLAY ATTITUDE
 MASK RCSFLAGS # ERRORS WAIT ATLEAST 4 MS FOR
 TS RCSFLAGS # RELAY CLOSURE
 TCF RETNMORE

NEEDLES3 CAF BIT6 # CHECK TO SEE IF IMU ERROR COUNTER
 EXTEND # IS ENABLED
 RAND CHAN12

```

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
|          | CCS    | A        | # IF NOT, RE-INITIALIZE NEEDLER.          |
|          | TCF    | NEEDLES  |                                           |
|          | CS     | RCSFLAGS | # SET UP INITIALIZATION FLAG IN RCSFLAGS. |
|          | MASK   | BIT3     |                                           |
|          | ADS    | RCSFLAGS |                                           |
|          | TCF    | RETNMORE |                                           |
| NEEDLES  | CAF    | TWO      |                                           |
| DACLOOP  | TS     | DINDX    |                                           |
|          | CS     | ONETENTH | # RESCALE INPUTS TO + OR - 1800 DEGREES.  |
|          | EXTEND |          |                                           |
|          | INDEX  | DINDX    |                                           |
|          | MP     | AK       |                                           |
|          | TS     | L        |                                           |
|          | CCS    | A        |                                           |
|          | CA     | DACLIMIT |                                           |
|          | TCF    | +2       |                                           |
|          | CS     | DACLIMIT |                                           |
|          | AD     | L        |                                           |
|          | TS     | T5TEMP   | # OVFLD CHK                               |
|          | TCF    | +4       |                                           |
|          | INDEX  | A        | # ON OVERFLOW LIMIT OUTPUT TO +-384       |
|          | CAF    | DACLIMIT |                                           |
|          | TS     | L        |                                           |
|          | INDEX  | DINDX    |                                           |
|          | CS     | EDRIVEX  | # CURRENT VALUE OF DAC                    |
|          | AD     | L        |                                           |
|          | INDEX  | DINDX    |                                           |
|          | ADS    | CDUXCMD  |                                           |
|          | INDEX  | DINDX    |                                           |
|          | LXCH   | EDRIVEX  |                                           |
|          | CCS    | DINDX    |                                           |
|          | TCF    | DACLOOP  |                                           |
|          | CAF    | 13,14,15 |                                           |
|          | EXTEND |          |                                           |
|          | WOR    | CHAN14   | # SET DAC ACTIVITY BITS                   |
|          | TCF    | RETNMORE |                                           |
|          | DEC    | -384     |                                           |
| DACLIMIT | DEC    | 16000    |                                           |
|          | DEC    | 384      |                                           |
| ONETENTH | OCT    | 03146    | # DECIMAL +0.1, SCALED AT 1.              |
| DSPLYALT | EQUALS | BIT4     | # 100 MS ALTERNATION BIT IN RCSFLAGS      |
| OVERSUB2 | TS     | 7        | # RETURNS A UNCHANGED OR LIMITED TO       |
|          | TC     | Q        | # POSMAX OR NEGMAX IF A HAS OVERFLOW      |
|          | INDEX  | A        |                                           |



|    |          |        |                      |    |
|----|----------|--------|----------------------|----|
| 1  |          |        |                      | 1  |
| 2  |          | CS     | LIMITS               | 2  |
| 3  |          | TC     | Q                    | 3  |
| 4  |          |        |                      | 4  |
| 5  | RETNMORE | EXTEND | # RETURN TO CHEKMORE | 5  |
| 6  |          | DCA    | MORECADR             | 6  |
| 7  |          | DTCB   |                      | 7  |
| 8  |          |        |                      | 8  |
| 9  |          | EBANK= | AOSQ                 | 9  |
| 10 | MORECADR | 2CADR  | CHEKMORE             | 10 |
| 11 |          |        |                      | 11 |
| 12 |          |        |                      | 12 |
| 13 |          |        |                      | 13 |
| 14 |          |        |                      | 14 |
| 15 |          |        |                      | 15 |
| 16 |          |        |                      | 16 |
| 17 |          |        |                      | 17 |
| 18 |          |        |                      | 18 |
| 19 |          |        |                      | 19 |
| 20 |          |        |                      | 20 |
| 21 |          |        |                      | 21 |
| 22 |          |        |                      | 22 |
| 23 |          |        |                      | 23 |
| 24 |          |        |                      | 24 |
| 25 |          |        |                      | 25 |
| 26 |          |        |                      | 26 |
| 27 |          |        |                      | 27 |
| 28 |          |        |                      | 28 |
| 29 |          |        |                      | 29 |
| 30 |          |        |                      | 30 |
| 31 |          |        |                      | 31 |
| 32 |          |        |                      | 32 |
| 33 |          |        |                      | 33 |
| 34 |          |        |                      | 34 |
| 35 |          |        |                      | 35 |
| 36 |          |        |                      | 36 |
| 37 |          |        |                      | 37 |
| 38 |          |        |                      | 38 |
| 39 |          |        |                      | 39 |
| 40 |          |        |                      | 40 |
| 41 |          |        |                      | 41 |
| 42 |          |        |                      | 42 |
| 43 |          |        |                      | 43 |
| 44 |          |        |                      | 44 |
| 45 |          |        |                      | 45 |
| 46 |          |        |                      | 46 |
| 47 |          |        |                      | 47 |
| 48 |          |        |                      | 48 |
| 49 |          |        |                      | 49 |
| 50 |          |        |                      | 50 |
| 51 |          |        |                      | 51 |
| 52 |          |        |                      | 52 |
| 53 |          |        |                      | 53 |
| 54 |          |        |                      | 54 |
| 55 |          |        |                      | 55 |
| 56 |          |        |                      | 56 |
| 57 |          |        |                      | 57 |
| 58 |          |        |                      | 58 |
| 59 |          |        |                      | 59 |
| 60 |          |        |                      | 60 |

```
1
2 BANK 16
3 SETLOC DAPS1
4 BANK
5
6 EBANK= PERROR
7 COUNT* $$/DAPP
8
9 # THE FOLLOWING T5RUPT ENTRY BEGINS THE PROGRAM WHICH CONTROLS THE P-AXIS ACTION OF THE LEM USING THE RCS JETS.
10 # THE NOMINAL TIME BETWEEN THE P-AXIS RUPTS IS 100 MS IN ALL NON-IDLING MODES OF THE DAP.
11
12 PAXIS CA MS100
13 ADS TIME5 # *** NECESSARY IN ORDER TO ALLOW
14 # SYNCHRONIZATION WITH OTHER INTERRUPTS ***
15
16 LXCH BANKRUPT # INTERRUPT LEAD IN (CONTINUED)
17 EXTEND
18 QXCH QRUPT
19
20 # CHECK IF DAP PASS IS PERMISSIBLE
21
22 CCS DAPZRPT # IF DAPZRPT POSITIVE, DAP (JASK) IS
23 TC BAILOUT # STILL IN PROGRESS AND A RESTART IS
24 OCT 02000 # CALLED FOR. IT IS NEVER ZERO
25
26 TC CHEKBITS # RETURN IS TC I+1 IF DAP SHOULD STAY ON.
27
28 CA CDUX # READ AND STORE CDU'S
29 TS DAPTREG4
30 CA CDUY
31 TS DAPTREG5
32 CA CDUZ
33 TS DAPTREG6
34
35 # ***** KALCMANU-DAP AND "RATE-HOLD"-DAP INTERFACE *****
36 #
37 # THE FOLLOWING SECTION IS EXECUTED EVERY 100 MS (10 TIMES A SECOND) WITHIN THE P-AXIS REACTION CONTROL SYSTEM
38 # AUTOPILOT (WHENEVER THE DAP IS IN OPERATION).
39
40 CA CDUXD
41 EXTEND
42 MSU DELCDUX
43 TC 1STOTWOS
44 TS CDUXD
45 CA CDUYD
46
47 EXTEND
48 MSU DELCDUY
49 TC 1STOTWOS
50 TS CDUYD
51 CA CDUZD
52 EXTEND
53 MSU DELCDUZ
54
55
56
57
58
59
60
```

```
1
2 TC 1STOTWOS
3 TS CDUZD
4 EXTEND
5 DIM TCP # DIMINISH MANUAL CONTROL DIRECT RATE
6 EXTEND # TIME COUNTERS.
7 DIM TCQR
8
9 # RATFLOOP COMPUTES JETRATEQ, JRATER, AND 1JACC*NO. PJETS IN ITEMPL.
10 # RETURNS TO BACKP.
11 #
12 # JETRATE = 1JACC*NO.PJETS*TJP (NOTE TJ IS THE TIME FIRED DURING CSP)
13 # JETRATEQ = 1JACCQ(TJU*NO.UJETS - TJV*NO.VJETS)
14 # JETRATER = 1JACCR(TJU*NO.UJETS + TJV*NO.VJETS)
15
16 1STOTWOS TCF PAXFILT # PROCEEDS TO RATELOOP AFTER SUPERJOB
17 CCS A
18 AD ONE
19
20 TC Q
21 CS A
22 TC Q
23 SUBDIVDE EXTEND # OVERFLOW PROTECTION ROUTINE TO GIVE
24 MP DAPTEMP3 # POSMAX OR NEGMAX IF THE DIVIDE WOULD
25 DAS OMEGAU # OVERFLOW
26
27 +3 EXTEND
28 DCA OMEGAU
29 DXCH DAPTEMP5
30 CCS OMEGAU
31 TCF +2
32
33 TCF DIVIDER
34 AD -OCT630
35 EXTEND
36 BZMF DIVIDER
37
38 CCS OMEGAU
39 CA POSMAX # 45 DEG/SEC
40 TC Q
41 CS POSMAX
42 TC Q
43
44 DIVIDER DXCH OMEGAU
45 EXTEND
46 DV DAPTREG4
47 TC Q
48
49 OVERSUB TS 7 # RETURNS A UNCHANGED OR LIMITED TO
50 TC Q # POSMAX OR NEGMAX IF A HAS OVERFLOW
51 INDEX A
52 CS BIT15 -1
```

```
1
2 TC Q
3
4 -OCT630 OCT 77147
5
6 BACKP CA DAPTEMP1
7 EXTEND
8 MP 1JACC
9 TS JETRATE
10
11 # BEGINNING OF THE RATE DERIVATION
12 # OMEGAP,Q,R BODY RATES SCALED AT PI/4
13 # TRAPEDP,Q,R BODY ANGLE ERRORS FROM PREDICTED ANGLE (PI/40)
14 # NP(QR)TRAPS NUMBER OF TIMES ANGLE ERROR HAS BEEN ACCUMULATED
15 # AOSQ(R)TERM CHANGE IN RATE DUE TO OFFSET ACCELERATION. (PI/4)
16 # JETRATE,Q,R CHANGE IN RATE DUE TO JET ACCELERATION. (PI/4)
17 # TRAPSIZE NEGATIVE LIMIT OF MAGNITUDE OF TRAPEDP, ETC.
18 # OMEGAU DP-TEMPORARY STORAGE
19 # OMEGA = OMEGA + JETRATE + AOSTERM (+TRAPED/NTRAPS IF TRAPED BIG)
20
21 CAE DAPTREG4 # CDUX IS STORED HERE
22 TS L
23 EXTEND
24 MSU OLDXFORP # SCALED AT PI
25 LXCH OLDXFORP
26 TS DAPTEMP1
27 CA 1/40
28 TS DAPTREG4
29 CS JETRATE
30 EXTEND
31 MP BIT14
32 ADS TRAPEDP
33 CA JETRATEQ
34 AD AOSQTERM
35 EXTEND
36 MP -BIT14
37 ADS TRAPEDQ
38 CA JETRATER
39 AD AOSRTERM
40 EXTEND
41 MP -BIT14
42 ADS TRAPEDR
43
44 CA DAPTREG5 # CDUY IS STORED HERE
45 TS L
46 EXTEND
47 MSU OLDYFORP # SCALED AT PI
48 LXCH OLDYFORP
49 TS DAPTEMP2
50 EXTEND
51 MP M11 # M11 SCALED AT 1
52
53
54
55
56
57
58
59
60
```



|    |                        |             |                                     |           |
|----|------------------------|-------------|-------------------------------------|-----------|
| 1  | # 1 AXIS ROSS ACTUATOR |             |                                     | PAGE 1121 |
| 2  | AD                     | DAPTEMP1    |                                     | 1         |
| 3  | DXCH                   | OMEGAU      |                                     | 2         |
| 4  |                        |             |                                     | 3         |
| 5  | TC                     | SUBDIVDE +3 | # RETURNS WITH CDU-RATE AT PI/4     | 4         |
| 6  |                        |             |                                     | 5         |
| 7  | EXTEND                 |             |                                     | 6         |
| 8  | SU                     | OMEGAP      |                                     | 7         |
| 9  | ADS                    | TRAPEDP     |                                     | 8         |
| 10 | TC                     | OVERSUB     |                                     | 9         |
| 11 | TS                     | TRAPEDP     |                                     | 10        |
| 12 | EXTEND                 |             |                                     | 11        |
| 13 | DCA                    | DAPTEMP5    |                                     | 12        |
| 14 | DAS                    | DXERROR     |                                     | 13        |
| 15 | CS                     | PLAST       |                                     | 14        |
| 16 | EXTEND                 |             |                                     | 15        |
| 17 | MP                     | 1/40        |                                     | 16        |
| 18 | DAS                    | DXERROR     | # MANUAL MODE X-ATTITUDE ERROR (DP) | 17        |
| 19 | CA                     | DAPTREG6    | # CDUZ IS STORED HERE               | 18        |
| 20 | TS                     | L           |                                     | 19        |
| 21 | EXTEND                 |             |                                     | 20        |
| 22 | MSU                    | OLDZFORQ    |                                     | 21        |
| 23 | TS                     | DAPTEMP3    |                                     | 22        |
| 24 | LXCH                   | OLDZFORQ    |                                     | 23        |
| 25 | CA                     | M21         |                                     | 24        |
| 26 | EXTEND                 |             |                                     | 25        |
| 27 | MP                     | DAPTEMP2    |                                     | 26        |
| 28 | DXCH                   | OMEGAU      |                                     | 27        |
| 29 | CA                     | M22         |                                     | 28        |
| 30 | TC                     | SUBDIVDE    |                                     | 29        |
| 31 |                        |             |                                     | 30        |
| 32 | EXTEND                 |             |                                     | 31        |
| 33 | SU                     | OMEGAQ      |                                     | 32        |
| 34 | ADS                    | TRAPEDQ     |                                     | 33        |
| 35 | TC                     | OVERSUB     |                                     | 34        |
| 36 | TS                     | TRAPEDQ     |                                     | 35        |
| 37 | EXTEND                 |             |                                     | 36        |
| 38 | DCA                    | DAPTEMP5    |                                     | 37        |
| 39 | DAS                    | DYERROR     |                                     | 38        |
| 40 | CS                     | QLAST       |                                     | 39        |
| 41 | EXTEND                 |             |                                     | 40        |
| 42 | MP                     | 1/40        |                                     | 41        |
| 43 | DAS                    | DYERROR     | # MANUAL MODE Y-ATTITUDE ERROR (DP) | 42        |
| 44 | CA                     | M31         |                                     | 43        |
| 45 | EXTEND                 |             |                                     | 44        |
| 46 | MP                     | DAPTEMP2    |                                     | 45        |
| 47 | DXCH                   | OMEGAU      |                                     | 46        |
| 48 | CA                     | M32         |                                     | 47        |
| 49 |                        |             |                                     | 48        |
| 50 | TC                     | SUBDIVDE    |                                     | 49        |
| 51 |                        |             |                                     | 50        |
| 52 |                        |             |                                     | 51        |
| 53 |                        |             |                                     | 52        |
| 54 |                        |             |                                     | 53        |
| 55 |                        |             |                                     | 54        |
| 56 |                        |             |                                     | 55        |
| 57 |                        |             |                                     | 56        |
| 58 |                        |             |                                     | 57        |
| 59 |                        |             |                                     | 58        |
| 60 |                        |             |                                     | 59        |

|    |                       |        |          |                                            |    |
|----|-----------------------|--------|----------|--------------------------------------------|----|
| 1  | # 1 AXIS RCS ACTUATOR |        |          | PAGE 1123                                  | 1  |
| 2  |                       | EXTEND |          |                                            | 2  |
| 3  |                       | SU     | OMEGAR   |                                            | 3  |
| 4  |                       | ADS    | TRAPEDR  |                                            | 4  |
| 5  |                       | TC     | OVERSUB  |                                            | 5  |
| 6  |                       | TS     | TRAPEDR  | # TRAPEDS HAVE ALL BEEN COMPUTED           | 6  |
| 7  |                       |        |          |                                            | 7  |
| 8  |                       | EXTEND |          |                                            | 8  |
| 9  |                       | DCA    | DAPTEMP5 |                                            | 9  |
| 10 |                       | DAS    | DZERROR  |                                            | 10 |
| 11 |                       | CS     | RLAST    |                                            | 11 |
| 12 |                       | EXTEND |          |                                            | 12 |
| 13 |                       | MP     | 1/40     |                                            | 13 |
| 14 |                       | DAS    | DZERROR  | # MANUAL MODE Z-ATTITUDE ERROR (DP)        | 14 |
| 15 |                       | CA     | DAPBOOLS | # PICK UP PAD LOADED STATE ESTIMATOR GAINS | 15 |
| 16 |                       | MASK   | CSMDOCKD |                                            | 16 |
| 17 |                       | EXTEND |          |                                            | 17 |
| 18 |                       | BZF    | LMONLY   |                                            | 18 |
| 19 |                       | EXTEND |          | # DOCKED                                   | 19 |
| 20 |                       | DCA    | DKOMEGAN |                                            | 20 |
| 21 |                       | DXCH   | DAPTREG4 |                                            | 21 |
| 22 |                       | CA     | DKTRAP   |                                            | 22 |
| 23 |                       | TCF    | +5       |                                            | 23 |
| 24 | LMONLY                | EXTEND |          | # UNDOCKED                                 | 24 |
| 25 |                       | DCA    | LMOMEGAN |                                            | 25 |
| 26 |                       | DXCH   | DAPTREG4 |                                            | 26 |
| 27 |                       | CA     | LMTRAP   |                                            | 27 |
| 28 | +5                    | TS     | DAPTREG6 |                                            | 28 |
| 29 |                       | CCS    | TRAPEDP  |                                            | 29 |
| 30 |                       | TCF    | +2       |                                            | 30 |
| 31 |                       | TCF    | SMALPDIF |                                            | 31 |
| 32 |                       | AD     | DAPTREG6 | # TRAPSIZE > ABOUT 77001 %-1.4DEG/SEC"     | 32 |
| 33 |                       | EXTEND |          |                                            | 33 |
| 34 |                       | BZMF   | SMALPDIF |                                            | 34 |
| 35 |                       | ZL     |          |                                            | 35 |
| 36 |                       | LXCH   | TRAPEDP  |                                            | 36 |
| 37 |                       | CA     | ZERO     |                                            | 37 |
| 38 |                       | EXTEND |          |                                            | 38 |
| 39 |                       | DV     | NPTRAPS  |                                            | 39 |
| 40 |                       | ADS    | OMEGAP   |                                            | 40 |
| 41 |                       | TC     | OVERSUB  |                                            | 41 |
| 42 |                       | TS     | OMEGAP   |                                            | 42 |
| 43 |                       | CA     | DAPTREG4 | ABOUT 10 OR 0 FOR DOCKED OR UNDOCKED       | 43 |
| 44 |                       | TS     | NPTRAPS  |                                            | 44 |
| 45 | SMALPDIF              | INCR   | NPTRAPS  |                                            | 45 |
| 46 | P-RATE                | CA     | JETRATE  |                                            | 46 |
| 47 |                       | ADS    | OMEGAP   |                                            | 47 |
| 48 |                       | TC     | OVERSUB  |                                            | 48 |
| 49 |                       | TS     | OMEGAP   |                                            | 49 |
| 50 |                       |        |          |                                            | 50 |
| 51 |                       | CCS    | TRAPEDQ  |                                            | 51 |
| 52 |                       |        |          |                                            | 52 |
| 53 |                       |        |          |                                            | 53 |
| 54 |                       |        |          |                                            | 54 |
| 55 |                       |        |          |                                            | 55 |
| 56 |                       |        |          |                                            | 56 |
| 57 |                       |        |          |                                            | 57 |
| 58 |                       |        |          |                                            | 58 |
| 59 |                       |        |          |                                            | 59 |
| 60 |                       |        |          |                                            | 60 |

|    |        |        |          |                                        |
|----|--------|--------|----------|----------------------------------------|
| 1  |        |        |          |                                        |
| 2  |        | TCF    | +2       |                                        |
| 3  |        | TCF    | Q-RATE   |                                        |
| 4  |        | AD     | DAPTREG6 | # TRAPSIZE > ABOUT 77001 %-1.4DEG/SEC" |
| 5  |        | EXTEND |          |                                        |
| 6  |        | BZMF   | Q-RATE   |                                        |
| 7  |        | ZL     |          |                                        |
| 8  |        | LXCH   | TRAPEDQ  |                                        |
| 9  |        | CA     | ZERO     |                                        |
| 10 |        | EXTEND |          |                                        |
| 11 |        | DV     | NQTRAPS  |                                        |
| 12 |        | TS     | DAPTEMP1 | # SAVE FOR OFFSET ESTIMATE             |
| 13 |        | ADS    | OMEGAQ   |                                        |
| 14 |        | TC     | OVERSUB  |                                        |
| 15 |        | TS     | OMEGAQ   |                                        |
| 16 |        | CA     | DAPTREG4 | # ABOUT 10 OR 0 FOR DOCKED OR UNDOCKED |
| 17 |        | XCH    | NQTRAPS  |                                        |
| 18 |        | AD     | DAPTREG5 | # KAOS > ABOUT 60D %N/N_60"            |
| 19 |        | XCH    | DAPTEMP1 |                                        |
| 20 |        | EXTEND |          |                                        |
| 21 |        | MP     | FIVE     |                                        |
| 22 |        | EXTEND |          |                                        |
| 23 |        | DV     | DAPTEMP1 |                                        |
| 24 |        | ADS    | AOSQ     |                                        |
| 25 | Q-RATE | INCR   | NQTRAPS  |                                        |
| 26 |        | CA     | JETRATEQ |                                        |
| 27 |        | AD     | AOSQTERM |                                        |
| 28 |        | ADS    | OMEGAQ   |                                        |
| 29 |        | TC     | OVERSUB  |                                        |
| 30 |        | TS     | OMEGAQ   |                                        |
| 31 |        |        |          |                                        |
| 32 |        | CCS    | TRAPEDR  |                                        |
| 33 |        | TCF    | +2       |                                        |
| 34 |        | TCF    | R-RATE   |                                        |
| 35 |        | AD     | DAPTREG6 | # TRAPSIZE > ABOUT 77001 %-1.4DEG/SEC" |
| 36 |        | EXTEND |          |                                        |
| 37 |        | BZMF   | R-RATE   |                                        |
| 38 |        | ZL     |          |                                        |
| 39 |        | LXCH   | TRAPEDR  |                                        |
| 40 |        | CA     | ZERO     |                                        |
| 41 |        | EXTEND |          |                                        |
| 42 |        | DV     | NRTRAPS  |                                        |
| 43 |        | TS     | DAPTEMP2 | # SAVE FOR OFFSET ESTIMATE             |
| 44 |        | ADS    | OMEGAR   |                                        |
| 45 |        | TC     | OVERSUB  |                                        |
| 46 |        | TS     | OMEGAR   |                                        |
| 47 |        | CA     | DAPTREG4 | # ABOUT 10 OR 0 FOR DOCKED OR UNDOCKED |
| 48 |        | XCH    | NRTRAPS  |                                        |
| 49 |        | AD     | DAPTREG5 | # KAOS > ABOUT 60D %N/N_60"            |
| 50 |        | XCH    | DAPTEMP2 |                                        |
| 51 |        | EXTEND |          |                                        |
| 52 |        |        |          |                                        |
| 53 |        |        |          |                                        |
| 54 |        |        |          |                                        |
| 55 |        |        |          |                                        |
| 56 |        |        |          |                                        |
| 57 |        |        |          |                                        |
| 58 |        |        |          |                                        |
| 59 |        |        |          |                                        |
| 60 |        |        |          |                                        |

```
1
2 MP FIVE
3 EXTEND
4 DV DAPTEMP2
5 ADS AOSR
6 R-RATE INCR NRTRAPS
7 CA JETRATER
8 AD AOSRTERM
9 ADS OMEGAR
10 TC OVERSUB
11 TS OMEGAR
12
13 # END OF RATE DERIVATION
14 # BEGIN OFFSET ESTIMATER
15 # IN POWERED FLIGHT, AOSTASK WILL BE CALLED EVERY 2 SECONDS.
16 # AOS = AOS + K*SUMRATE
17
18 CS DAPBOOLS
19 MASK DRIFTBIT
20 CCS A
21 TCF WORKTIME
22 TS ALPHAQ # ZERO THE OFFSET ACCELERATION VALUES.
23 TS ALPHAR
24 TS AOSQTERM
25 TS AOSRTERM
26 TS AOSQ
27 TS AOSR
28 TCF PRETIMCK
29 KAOS DEC 60
30 WORKTIME CA QACCDOT
31 EXTEND
32 MP CALLCODE # OCTAL 00032 IS DECIMAL .1 AT 2(6).
33 DAS AOSQ
34 CA AOSQ
35 TS ALPHAQ
36 EXTEND
37 MP 200MS # .2 AT 1
38 TS AOSQTERM
39 CA RACCDOT
40 EXTEND
41 MP CALLCODE # OCTAL 00032 IS DECIMAL .1 AT 2(6).
42 DAS AOSR
43 CA AOSR
44 TS ALPHAR
45 EXTEND
46 MP 200MS # .2 AT 1
47 TS AOSRTERM
48 TCF PRETIMCK
```

```
1 PAXFILT CA CALLGMBL # EXECUTE ACDT+C12, IF NEEDED.
2 MASK RCSFLAGS
3
4 CCS A # CALLGMBL IS NOT BIT15, SO THIS TEST IS
5 TC ACDT+C12 # VALID.
6
7 DXCH ARUPT
8 DXCH DAPARUPT
9 CA SUPERJOB # SETTING UP THE SUPERJOB
10 XCH BRUPT
11 LXCH QRUPT
12 DXCH DAPBQRPT
13 CA SUPERADR
14 DXCH ZRUPT
15 DXCH DAPZRUPT
16 TCF NOQBRSM +1 # RELINT (JUST IN CASE) AND RESUME, IN THE
17 # FORM OF A JASK, AT SUPERJOB.
18
19 SUPERADR GENADR SUPERJOB +1
20
21 # COUNT DOWN GIMBAL DRIVE TIMERS AND TURN OFF DRIVES IF REQUIRED.
22
23 SUPERJOB TCF RATELOOP
24 PRETIMCK CCS QGIMTIMR
25
26 TCF DECQTIMR # POSITIVE -- COUNTING DOWN
27 CHKRTIMR TCF TURNOFFQ # NEGATIVE -- DRIVE SHOULD BE ENDED
28 CCS RGIMTIMR # NEGATIVE -- INACTIVE
29 TCF DECRTIMR # (NEG ZERO -- IMPOSSIBLE)
30 TCF TURNOFFR # REPEATED (ABOVE) FOR R AXIS.
31
32 EXTEND # DECREMENT DOCKED JET INHIBITION COUNTERS
33 DIM PJETCTR
34 EXTEND
35 DIM UJETCTR
36 EXTEND
37 DIM VJETCTR
38 CA BIT12
39 MASK RCSFLAGS
40 EXTEND
41 BZF SKIPPAXS
42 TC CHKVISFZ
43 TS QGIMTIMR # COUNT TIMERS DOWN TO POS ZERO.
44
45 TCF CHKRTIMR
46 TS RGIMTIMR
47 TCF CHKRTIMR +3
48
49 TURNOFFQ TS NEGUQ # HALT DRIVES.
50 TS QACCDOT
51 CS QGIMBITS
52 EXTEND
53
54
55
56
57
58
59
60
```

```
WAND CHAN12
CAF NEGMAX

TS QGIMTIMR
TCF CHKRTIMR
TS NEGUR

TS RACCDOT
CS RGIMBITS
EXTEND

WAND CHAN12
CAF NEGMAX
TS RGIMTIMR
TCF CHKRTIMR +3
QGIMBITS EQUALS OCT1400 # BITS 9 AND 10 (OF CHANNEL 12).
RGIMBITS EQUALS PRI06 # BITS 11 AND 12 (OF CHANNEL 12).

SKIPPAXS CS RCSFLAGS
 MASK BIT12
 ADS RCSFLAGS # BIT 12 SET TO 1.
 TCF QRAXIS # GO TO QRAXIS OR TO CTS.

Y-X TRANSLATION
#
INPUT: BITS 9-12 OF CH31 (FROM TRANSLATION CONTROLLER)
#
OUTPUT: NEXTP
#
NEXTP IS THE CHANNEL 6 CODE OF JETS FOR THE DESIRED TRANSLATION.
IF THERE ARE FAILURES IN THE DESIRED POLICY, THEN
(1) FOR DIAGONAL TRANS: UNFAILED PAIR
ALARM (IF NO PAIR)
(2) FOR PRINCIPAL TRANS: TRY TO TACK WITH DIAGONAL PAIRS
ALARM (IF DIAGONAL PAIRS ARE FAILED)

CHKVISFZ EXTEND
 READ CHAN31
 CS A
 MASK 07400OCT
 EXTEND
 BZF TSNEXTP
 EXTEND
 MP BIT7
 INDEX A
 CA INDXYZ
 TS ROTINDEX

TRYUORV CA SIX
 TC SELECTYZ
 CS SIX
 AD NUMBERT
 EXTEND
```

|               |                    |                     |                                          |
|---------------|--------------------|---------------------|------------------------------------------|
|               | BZF                | TSNEXTP -1          |                                          |
|               | CS                 | FIVE                |                                          |
|               | AD                 | ROTINDEX            |                                          |
|               | EXTEND             |                     |                                          |
|               | BZMF               | ALTERYZ             |                                          |
|               | CS                 | NUMBERT             |                                          |
|               | AD                 | FOUR                |                                          |
|               | EXTEND             |                     |                                          |
| ABORTYZ       | BZMF               | TSNEXTP -1          |                                          |
|               | TC                 | ALARM               |                                          |
|               | OCT                | 02001               |                                          |
|               | CA                 | BIT1                | # INVERT BIT 1 OF RCSFLAGS.              |
|               | LXCH               | RCSFLAGS            |                                          |
|               | EXTEND             |                     |                                          |
|               | RXOR               | 1                   |                                          |
|               | TS                 | RCSFLAGS            |                                          |
|               | CA                 | ZERO                |                                          |
| ALTERYZ       | TCF                | TSNEXTP             |                                          |
|               | CA                 | BIT1                | # INVERT BIT 1 OF RCSFLAGS.              |
|               | LXCH               | RCSFLAGS            |                                          |
|               | EXTEND             |                     |                                          |
|               | RXOR               | 1                   |                                          |
|               | TS                 | RCSFLAGS            |                                          |
|               | MASK               | BIT1                |                                          |
|               | AD                 | FOUR                |                                          |
|               | ADS                | ROTINDEX            |                                          |
|               | TCF                | TRYUORV             |                                          |
|               | CA                 | POLYTEMP            |                                          |
| TSNEXTP       | TS                 | NEXTP               |                                          |
| # STATE LOGIC |                    |                     |                                          |
| #             | CHECK IN ORDER:    | IF ON               |                                          |
| #             | LPDPHASE           | GO TO PURGENCY      |                                          |
| #             | PULSES             | MINIMUM PULSE LOTIC |                                          |
| #             | DETENT(BIT15 CH31) | RATE COMMAND        |                                          |
| #             | GOTO TO PURGENCY   |                     |                                          |
|               | CA                 | BIT13               | # CHECK STICK IF IN ATT. HOLD.           |
|               | EXTEND             |                     |                                          |
|               | RAND               | CHAN31              |                                          |
|               | EXTEND             |                     |                                          |
|               | BZF                | MANMODE             |                                          |
|               | CA                 | DAPBOOLS            |                                          |
|               | MASK               | XOVINHIB            |                                          |
|               | CCS                | A                   |                                          |
|               | TCF                | PURGENCY            | # ATTITUDE STEER DURING VISIBILITY PHASE |
|               | TCF                | DETENTCK            |                                          |
| MANMODE       | CA                 | PULSES              | # PULSES IS ONE FOR PULSE MODE           |
|               | MASK               | DAPBOOLS            |                                          |

|    |        |                                                                                                         |             |                                      |    |
|----|--------|---------------------------------------------------------------------------------------------------------|-------------|--------------------------------------|----|
| 2  |        | EXTEND                                                                                                  |             |                                      | 2  |
| 3  |        | BZF                                                                                                     | DETENTCK    | # BRANCH FOR RATE COMMAND            | 3  |
| 4  |        |                                                                                                         |             |                                      | 4  |
| 5  |        | CA                                                                                                      | ZERO        |                                      | 5  |
| 6  |        | TS                                                                                                      | PERROR      |                                      | 6  |
| 7  |        |                                                                                                         |             |                                      | 7  |
| 8  | #      | MINIMUM IMPULSE MODE                                                                                    |             |                                      | 8  |
| 9  |        |                                                                                                         |             |                                      | 9  |
| 10 |        | CA                                                                                                      | CDUX        |                                      | 10 |
| 11 |        | TS                                                                                                      | CDUXD       |                                      | 11 |
| 12 |        |                                                                                                         |             |                                      | 12 |
| 13 |        | CCS                                                                                                     | OLDPMIN     |                                      | 13 |
| 14 |        | TCF                                                                                                     | CHECKP      |                                      | 14 |
| 15 |        |                                                                                                         |             |                                      | 15 |
| 16 | FIREP  | CA                                                                                                      | BIT3        |                                      | 16 |
| 17 |        | EXTEND                                                                                                  |             |                                      | 17 |
| 18 |        | RAND                                                                                                    | CHAN31      |                                      | 18 |
| 19 |        | EXTEND                                                                                                  |             |                                      | 19 |
| 20 |        | BZF                                                                                                     | +XMIN       |                                      | 20 |
| 21 |        |                                                                                                         |             |                                      | 21 |
| 22 |        | CA                                                                                                      | BIT4        |                                      | 22 |
| 23 |        | EXTEND                                                                                                  |             |                                      | 23 |
| 24 |        | RAND                                                                                                    | CHAN31      |                                      | 24 |
| 25 |        | EXTEND                                                                                                  |             |                                      | 25 |
| 26 |        | BZF                                                                                                     | -XMIN       |                                      | 26 |
| 27 |        |                                                                                                         |             |                                      | 27 |
| 28 |        | TCF                                                                                                     | JETSOFF     |                                      | 28 |
| 29 |        |                                                                                                         |             |                                      | 29 |
| 30 | CHECKP | EXTEND                                                                                                  |             |                                      | 30 |
| 31 |        | READ                                                                                                    | CHAN31      |                                      | 31 |
| 32 |        | CS                                                                                                      | A           |                                      | 32 |
| 33 |        | MASK                                                                                                    | OCT14       |                                      | 33 |
| 34 |        | TS                                                                                                      | OLDPMIN     |                                      | 34 |
| 35 |        | TCF                                                                                                     | JETSOFF     |                                      | 35 |
| 36 |        |                                                                                                         |             |                                      | 36 |
| 37 | -XMIN  | CS                                                                                                      | TEN         | # ANYTHING LESS THAN 14MS. CORRECTED | 37 |
| 38 |        | TCF                                                                                                     | +2          | # IN JET SELECTION ROUTINE           | 38 |
| 39 | +XMIN  | CA                                                                                                      | TEN         |                                      | 39 |
| 40 |        | TS                                                                                                      | TJP         |                                      | 40 |
| 41 |        | CA                                                                                                      | ONE         |                                      | 41 |
| 42 |        | TS                                                                                                      | OLDPMIN     |                                      | 42 |
| 43 |        | TCF                                                                                                     | PJETSLEC -6 |                                      | 43 |
| 44 |        |                                                                                                         |             |                                      | 44 |
| 45 | #      | MANUAL RATE COMMAND MODE                                                                                |             |                                      | 45 |
| 46 | #      | =====                                                                                                   |             |                                      | 46 |
| 47 | #      | BY ROBERT F. STENGEL                                                                                    |             |                                      | 47 |
| 48 | #      |                                                                                                         |             |                                      | 48 |
| 49 | #      | THIS MODE PROVIDES RCAH MANUAL CONTROL THRU 2 CONTROL LAWS: 1) DIRECT RATE AND 2) PSEUDO-AUTO.          |             |                                      | 49 |
| 50 | #      | THE DIRECT RATE MODE AFFORDS IMMEDIATE CONTROL WITHOUT OVERSHOOT. THE PSEUDO-AUTO MODE PROVIDES PRECISE |             |                                      | 50 |
| 51 | #      | RATE CONTROL AND ATTITUDE HOLD.                                                                         |             |                                      | 51 |
| 52 | #      |                                                                                                         |             |                                      | 52 |
| 53 |        |                                                                                                         |             |                                      | 53 |
| 54 |        |                                                                                                         |             |                                      | 54 |
| 55 |        |                                                                                                         |             |                                      | 55 |
| 56 |        |                                                                                                         |             |                                      | 56 |
| 57 |        |                                                                                                         |             |                                      | 57 |
| 58 |        |                                                                                                         |             |                                      | 58 |
| 59 |        |                                                                                                         |             |                                      | 59 |
| 60 |        |                                                                                                         |             |                                      | 60 |



```
1 # IN DIRECT RATE, JETS ARE FIRED WHEN STICK POSITION CHANGES BY A FIXED NUMBER OF INCREMENTS IN ONE DAP CYCLE.
2 # THE 'BREAKOUT LEVEL' IS .6 D/S FOR LM-ONLY AND .3 D/S FOR CSM-DOCKED. THIS LAW NULLS THE RATE ERROR TO WITHIN
3 # THE 'TARGET DEADBAND', WHICH EQUALS THE BREAKOUT LEVEL.
4 #
5 # IN PSEUDO-AUTO, BODY-FIXED RATE AND ATTITUDE ERRORS ARE SUPPLIED TO TJETLAW, WHICH EXERCISES CONTROL.
6 # CONTROL SWITCHES FROM DIRECT RATE TO PSEUDO-AUTO IF THE TARGET DB IS ACHIEVED OR IF TIME IN (1) EXCEEDS 4 SEC.
7 # IF THE INITIAL COMMAND DOES NOT EXCEED THE BREAKOUT LEVEL, CONTROL GOES TO PSEUDO-AUTO IMMEDIATELY.
8 #
9 # SINCE P-AXIS CONTROL IS SEPARATE FROM Q,R AXES CONTROL, IT IS POSSIBLE TO USE (1) IN P-AXIS AND (2) IN Q,R AXES,
10 # OR VICE VERSA. THIS ALLOWS A DEGREE OF ATTITUDE HOLD IN UNCONTROLLED AXES. DUE TO U,V CONTROL, HOWEVER, Q AND
11 # R AXES ARE COUPLED AND MUST USE THE SAME CONTROL LAW.
12 #
13 # HAND CONTROLLER COMMANDS ARE SCALED BY A LINEAR/QUADRATIC LAW. FOR THE LM-ALONE, MAXIMUM COMMANDED RATES ARE 20
14 # AND 4 D/S IN NORMAL AND FINE SCALING; HOWEVER, STICK SENSITIVITY AT ZERO COUNTS (OBTAINED AT A STICK DEFLECTION
15 # OF 2 DEGREES FROM THE CENTERED POSITION) IS .5 OR .1 D/S PER DEGREE. NORMAL AND FINE SCALINGS FOR THE CSM-DOCKED
16 # CASE IS AUTOMATICALLY SET TO 1/10 THE ABOVE VALUES. SCALING IS DETERMINED IN ROUTINE 3.
17 #
18 # ZEROENBL ENABLES COUNTERS SO THEY CAN BE READ NEXT TIME
19 # JUSTOUT FIRST DETECTION OF OUT OF DETENT (BY OURRCBIT)
20
21 DETENTCK EXTEND
22 READ CHAN31
23 TS CH31TEMP
24
25 MASK BIT15 # CHECK OUT-OF-DETENT BIT.
26 EXTEND
27 BZF RHCMOVED # BRANCH IF OUT OF DETENT.
28 CAF OURRCBIT # IN DETENT. CHECK THE RATE COMMAND BIT.
29 MASK DAPBOOLS
30 EXTEND
31 BZF PURGENCY # BRANCH IF NOT IN RATE COMMAND LAST PASS.
32
33 #
34
35 CA BIT9 # JUST IN DETENT??
36 MASK RCSFLAGS
37
38 EXTEND
39 BZF RUTH
40 CAF BIT13 # CHECK FOR ATTITUDE HOLD.
41
42 EXTEND
43 RAND CHAN31
44 EXTEND
45 BZF RATEDAMP # BRANCH IF IN ATTITUDE HOLD.
46
47 CS BITS9,11 # IN AUTO.
48 MASK RCSFLAGS # (X-AXIS OVERRIDE)
49 TS RCSFLAGS # ZERO ORBIT (BIT 11) AND JUST-IN BIT (9).
50 TCF RATEDAMP
51
52 RUTH CA RCSFLAGS
53 MASK PBIT # IN ATTITUDE HOLD.
54 EXTEND
55 BZF +2
56 TCF RATEDAMP # BRANCH IF P-RATE DAMPING IS FINISHED.
57
58
59
60
```

```
1
2 CA RCSFLAGS
3 MASK QRBIT
4 EXTEND
5 BZF RATEDONE # BRANCH IF Q,R RATE DAMPING IS FINISHED.
6 TCF RATEDAMP
```

```
7
8 # =====
```

```
9
10 1/10SEC OCT 1
11 40CYC OCT 50
12 PQRBIT OCT 74777
13 BITS9,11 EQUALS EBANK5
14 LINRATP DEC 46
```

```
15
16 # =====
```

```
17
18 RATEDONE CS OURRCBIT # MANUAL COMMAND AND DAMPING COMPLETED IN
19 # ALL AXES.
20 INHINT
21 MASK DAPBOOLS
22 TS DAPBOOLS
```

```
23
24 # READ CDUS INTO CDU DESIRED REGISTERS
```

```
25
26 CAF BIT13
27 EXTEND
28 RAND CHAN31
29 EXTEND
30 BZF +4
31 CA CDUX # (X-AXIS OVERRIDE)
32 TS CDUXD
33 TC +3
34 TC IBNKCALL
35 FCADR ZATTEROR
36 RELINT
37 TCF PURGENCY
```

```
38
39 JUSTOUT TS PERROR
40 # INITIALIZATION -- FIRST MANUAL PASS.
41 CA OURRCBIT
```

```
42 ADS DAPBOOLS
43 CA ZERO
44 TS DXERROR
45 TS DXERROR +1
46 TS DYERROR
47 TS DYERROR +1
48 TS DZERROR
49 TS DZERROR +1
50 TS PLAST
51 TS QLAST
52 TS RLAST
53 TS Q-RHCCTR
54 CA R-RHCCTR
55 MASK PQRBIT
56 TS RCSFLAGS
57 # BITS 10 AND 11 OF RCSFLAGS ARE 0.
```

|          |        |            |                                            |
|----------|--------|------------|--------------------------------------------|
|          | CS     | RCSFLAGS   | # SET 'JUST-IN' BIT TO 1.                  |
|          | MASK   | BIT9       |                                            |
|          | ADS    | RCSFLAGS   |                                            |
|          | TC     | ZEROENBL   |                                            |
|          | TCF    | JETSOFF    |                                            |
| ZEROENBL | LXCH   | R-RHCCTR   |                                            |
|          | CA     | Q-RHCCTR   |                                            |
|          | DXCH   | SAVEHAND   |                                            |
|          | CA     | ZERO       |                                            |
|          | TS     | P-RHCCTR   |                                            |
|          | TS     | Q-RHCCTR   |                                            |
|          | TS     | R-RHCCTR   |                                            |
|          | CA     | BITS8,9    |                                            |
|          | EXTEND |            |                                            |
|          | WOR    | CHAN13     | # COUNTERS ZEROED AND ENABLED              |
|          | TC     | Q          |                                            |
| RATEDAMP | CA     | ZERO       |                                            |
|          | TS     | P-RHCCTR   |                                            |
|          | TCF    | RATERROR   |                                            |
| RHCMOVED | CA     | OURRCBIT   | # P CONTROL                                |
|          | MASK   | DAPBOOLS   |                                            |
|          | EXTEND |            |                                            |
|          | BZF    | JUSTOUT -1 |                                            |
| RATERROR | CA     | CDUX       | # FINDCDUW REQUIRES THAT CDUXD=CDUX DURING |
|          | TS     | CDUXD      | # X-AXIS OVERRIDE                          |
|          | CCS    | P-RHCCTR   |                                            |
|          | TCF    | +3         |                                            |
|          | TCF    | +2         |                                            |
|          | TCF    | +1         |                                            |
|          | DOUBLE |            | # LINEAR/QUADRATIC CONTROLLER SCALING      |
|          | DOUBLE |            | # (SEE EXPLANATION OF Q,R-AXES RCS         |
|          | AD     | LINRATP    | # AUTOPILOT)                               |
|          | EXTEND |            |                                            |
|          | MP     | P-RHCCTR   |                                            |
|          | CA     | L          |                                            |
|          | EXTEND |            |                                            |
|          | MP     | STIKSENS   |                                            |
|          | XCH    | PLAST      |                                            |
|          | COM    |            |                                            |
|          | AD     | PLAST      |                                            |
|          | TS     | DAPTEMP1   |                                            |
|          | TC     | ZEROENBL   | # INTERVAL. ZERO AND ENABLE ACA COUNTERS.  |
|          | CS     | PLAST      |                                            |
|          | AD     | OMEGAP     |                                            |
|          | TS     | EDOTP      |                                            |
|          | CCS    | DAPTEMP1   | # IF P COMMAND CHANGE EXCEEDS BREAKOUT     |
|          | TCF    | +3         | # LEVEL, GO TO DIRECT RATE CONTROL. IF NOT |
|          | TCF    | +8D        | # CHECK FOR DIRECT RATE CONTROL LAST TIME. |
|          | TCF    | +1         |                                            |

|    |      |        |             |                                            |
|----|------|--------|-------------|--------------------------------------------|
| 1  |      |        |             |                                            |
| 2  |      | AD     | -RATEDB     |                                            |
| 3  |      | EXTEND |             |                                            |
| 4  |      | BZMF   | +4          |                                            |
| 5  |      | CA     | 40CYC       |                                            |
| 6  |      | TS     | TCP         |                                            |
| 7  |      | TC     | PEGI        |                                            |
| 8  |      | CA     | RCSFLAGS    | # CHECK FOR DIRECT RATE COMMAND LAST TIME. |
| 9  |      | MASK   | PBIT        |                                            |
| 10 |      | EXTEND |             |                                            |
| 11 |      | BZF    | +2          |                                            |
| 12 |      | TC     | PEGI        | # TO PURE RATE COMMAND                     |
| 13 |      | CA     | DXERROR     | # PSEUDO-AUTO CONTROL.                     |
| 14 |      | TS     | E           | # X-ATTITUDE ERROR (SP)                    |
| 15 |      | TS     | PERROR      | # LOAD P-AXIS ERROR FOR MODEL FDAI DISPLAY |
| 16 |      | TC     | PURGENCY +4 |                                            |
| 17 | PEGI | CA     | CDUX        | # DIRECT RATE CONTROL.                     |
| 18 |      | TS     | CDUXD       |                                            |
| 19 |      | CA     | ZERO        |                                            |
| 20 |      | TS     | DXERROR     |                                            |
| 21 |      | TS     | DXERROR +1  |                                            |
| 22 |      | TS     | PERROR      | # ZERO P-AXIS ERROR FOR MODEL FDAI DISPLAY |
| 23 |      | CCS    | EDOTP       |                                            |
| 24 |      | TC     | +3          |                                            |
| 25 |      | TC     | +2          |                                            |
| 26 |      | TC     | +1          |                                            |
| 27 |      | TS     | ABSEDOTP    |                                            |
| 28 |      | AD     | TARGETDB    |                                            |
| 29 |      | EXTEND |             | # IF RATE ERROR IS LESS THAN DEADBANK,     |
| 30 |      | BZMF   | LAST        | # FIRE, AN SWITCH TO PSEUDO-AUTO.          |
| 31 |      | CA     | TCP         |                                            |
| 32 |      | EXTEND |             | # IF TIME IN RATE COMMAND EXCEEDS 4 SEC.   |
| 33 |      | BZMF   | LAST        |                                            |
| 34 |      | CS     | RCSFLAGS    |                                            |
| 35 |      | MASK   | PBIT        |                                            |
| 36 |      | ADS    | RCSFLAGS    | # BIT 10 IS 1.                             |
| 37 |      | TCF    | +4          |                                            |
| 38 | LAST | CS     | PBIT        |                                            |
| 39 |      | MASK   | RCSFLAGS    |                                            |
| 40 |      | TS     | RCSFLAGS    | # BIT 10 IS 0.                             |
| 41 |      | CS     | EDOTP       |                                            |
| 42 |      | EXTEND |             |                                            |
| 43 |      | MP     | 1/ANETP     | # 1/2JTACC SCALED AT 2EXP(7)/PI            |
| 44 |      | DAS    | A           |                                            |
| 45 |      | TC     | OVERSUB     |                                            |
| 46 |      | EXTEND |             |                                            |
| 47 |      | MP     | 25/32       | # A CONTAINS TJET SCALED AT 2EXP(4)(16/25) |
| 48 |      | TS     | TJP         | # 4.JET TIME                               |
| 49 |      | CA     | ABSEDOTP    |                                            |
| 50 |      | AD     | -2JETLIM    | # COMPARING DELTA RATE WITH 2 JET LIMIT    |
| 51 |      | EXTEND |             |                                            |
| 52 |      |        |             |                                            |
| 53 |      |        |             |                                            |
| 54 |      |        |             |                                            |
| 55 |      |        |             |                                            |
| 56 |      |        |             |                                            |
| 57 |      |        |             |                                            |
| 58 |      |        |             |                                            |
| 59 |      |        |             |                                            |
| 60 |      |        |             |                                            |

```
1
2 BZMF +3
3
4 CA SIX
5 TCF +8D
6 CA TJP
7 ADS TJP
8
9 # GOES TO PJETSLEC FOR TWO JETS
10 # P-JET-SELECTION-ROUTINE (ROTATION)
11 #
12 # INPUT: NUMBERT 4,5,6 FOR WHICH PAIR OR 4 JETS
13 # TJP + FOR +P ROTATION
14 #
15 # OUTPUT: CHANNEL 6
16 # PJUMPADR FOR P-AXIS SKIP
17 # (JTLST CALL) (SMALL TJP)
18 #
19 # ORDER OF POLICIES TRIED IN CASE OF FAILURE.
20 # +P -P
21 # 7,15 8,16
22 # 4,12 3,11
23 # 4,7 8,11
24 # 7,12 11,16
25 # 12,15 3,16
26 # 4,15 3,8
27 # ALARM ALARM
28
29 CA AORBSYST
30 MASK DAPBOOLS
31
32 CCS A
33 CA ONE
34 AD FOUR
35 PJETSLEC TS NUMBERT
36 CA ONE
37 TS L
38 CCS TJP
39 TCF +5
40 TCF JETSOFF
41 TCF +2
42 TCF JETSOFF
43 ZL
44 AD ONE
45 TS ABSTJ
46 LXCH ROTINDEX
47 TC SELECTP
48 CS SIX
49 AD NUMBERT
50 EXTEND
51 BZF +2
52
53 CS TWO
```

```
1
2 AD FOUR
3 TS NO.PJETS
4 CA POLYTEMP
5 TC WRITEP
6 CS ABSTJ
7 AD +150MST6
8 EXTEND
9 BZMF QRAXIS # GO TO QRAXIS OR TO GTS.
10
11 AD -136MST6
12 EXTEND
13 BZMF +5
14
15 ADS ABSTJ
16 INDEX ROTINDEX
17 CA MINTIMES
18 TS TJP
19
20 CA ABSTJ
21 ZL
22 INHINT
23 DXCH T6FURTHA
24 TC IBNKCALL
25 CADR JTLST
26 CS BIT12
27 MASK RCSFLAGS
28 TS RCSFLAGS # BIT 12 SET TO 0.
29 TC ALTSYST
30 TCF QRAXIS
31
32 ALTSYST CA DAPBOOLS # ALTERNATE P-AXIS JETS
33 TS L
34 CA AORBSYST
35 EXTEND
36 RXOR LCHAN
37 TS DAPBOOLS
38 RELINT
39 TC Q
40
41 DKALT TC ALTSYST
42
43 JETSOFF TC WRITEP -1
44 CA ZERO
45 TS TJP
46 TCF QRAXIS
47
48 # (NOTE -- M13 = 1 IDENTICALLY IMPLIES NULL MULTIPLICATION.)
49
50 CALCPERR CA CDUY # P-ERROR CALCULATION.
51 EXTEND
52 MSU CDUYD # CDU VALUE -- ANGLE DESIRED (Y-AXIS)
53
54
55
56
57
```

```
1
2 EXTEND
3 MP M11 # (CDUY-CDUYD)M11 SCALED AT PI RADIAN
4 XCH E # SAVE FIRST TERM (OF TWO)
5 CA CDUX # THIRD COMPONENT
6 EXTEND
7 MSU CDUXD # CDU VALUE -- ANGLE DESIRED (X-AXIS)
8 #
9 # EXTEND
10 MP M13
11 AD DELPEROR # KALCMANU INTERFACE ERROR.
12 ADS E # SAVE SUM OF TERMS. COULD BE OVERFLOW.
13 XCH PERROR # SAVE P-ERROR FOR EIGHT-BALL DISPLAY.
14 TC Q # RETURN TO CALLER
15
16 # P-AXIS URGENCY FUNCTION CALCULATION.
17
18 PURGENCY TC CALCPERR # CALCULATE P-AXIS ERRORS.
19 CS OMEGAPD # THIS CODING IS COMMON TO BOTH LM DAP AND
20 AD OMEGAP # SPS-BACKUP MODE.
21 TS EDOTP # EDOTP = OMEGAP - OMEGAPD AT PI/4 RAD/SEC
22
23 CS ONE
24 TS AXISCTR
25 CA DAPBOOLS
26 MASK CSMDOCKD
27 EXTEND
28 BZF HEADTJET
29 INHINT
30 TC IBNKCALL # IF CSMDOCKD = 1, GOT TO DOCKED RCS LOGIC
31 CADR SPSRCS
32
33 CA TJP
34 EXTEND
35 BZF DKALT # IF TJP = ZERO, CHANGE AORBSYST.
36 RELINT
37 TCF PJETSLEC -6 # SELECT AORBSYST AND USE TWO JETS.
38 HEADTJET CA ZERO
39 TS SENSETYP
40 INHINT
41 TC IBNKCALL
42 CADR TJETLAW
43 RELINT
44
45 CS FIREFCT
46 AD -FOURDEG
47 EXTEND
48 BZMF PJETSLEC -6
49 CCS TJP
50 TCF +2
51 TCF JETSOFF
```



|    |          |        |             |    |
|----|----------|--------|-------------|----|
| 1  |          |        |             | 1  |
| 2  |          | AD     | -160MST6    | 2  |
| 3  |          | EXTEND |             | 3  |
| 4  |          | BZMF   | PJETSLEC -6 | 4  |
| 5  |          | CA     | SIX         | 5  |
| 6  |          | TCF    | PJETSLEC -1 | 6  |
| 7  | -160MST6 | DEC    | -256        | 7  |
| 8  | -FOURDEG | DEC    | -.08888     | 8  |
| 9  |          |        |             | 9  |
| 10 |          |        |             | 10 |
| 11 |          |        |             | 11 |
| 12 |          |        |             | 12 |
| 13 |          |        |             | 13 |
| 14 |          |        |             | 14 |
| 15 |          |        |             | 15 |
| 16 |          |        |             | 16 |
| 17 |          |        |             | 17 |
| 18 |          |        |             | 18 |
| 19 |          |        |             | 19 |
| 20 |          |        |             | 20 |
| 21 |          |        |             | 21 |
| 22 |          |        |             | 22 |
| 23 |          |        |             | 23 |
| 24 |          |        |             | 24 |
| 25 |          |        |             | 25 |
| 26 |          |        |             | 26 |
| 27 |          |        |             | 27 |
| 28 |          |        |             | 28 |
| 29 |          |        |             | 29 |
| 30 |          |        |             | 30 |
| 31 |          |        |             | 31 |
| 32 |          |        |             | 32 |
| 33 |          |        |             | 33 |
| 34 |          |        |             | 34 |
| 35 |          |        |             | 35 |
| 36 |          |        |             | 36 |
| 37 |          |        |             | 37 |
| 38 |          |        |             | 38 |
| 39 |          |        |             | 39 |
| 40 |          |        |             | 40 |
| 41 |          |        |             | 41 |
| 42 |          |        |             | 42 |
| 43 |          |        |             | 43 |
| 44 |          |        |             | 44 |
| 45 |          |        |             | 45 |
| 46 |          |        |             | 46 |
| 47 |          |        |             | 47 |
| 48 |          |        |             | 48 |
| 49 |          |        |             | 49 |
| 50 |          |        |             | 50 |
| 51 |          |        |             | 51 |
| 52 |          |        |             | 52 |
| 53 |          |        |             | 53 |
| 54 |          |        |             | 54 |
| 55 |          |        |             | 55 |
| 56 |          |        |             | 56 |
| 57 |          |        |             | 57 |
| 58 |          |        |             | 58 |
| 59 |          |        |             | 59 |
| 60 |          |        |             | 60 |



# JET POLICY CONSTRUCTION SUBROUTINE

#

# INPUT: ROTINDEX, NUMBERT

#

# OUTPUT: POLYTEMP (JET POLICY)

#

# THIS SUBROUTINE SELECT A SUBSET OF THE DESIRED JETS WHICH HAS NO FAILURE

SELECTP

CA  
TS  
INDEX

SIX  
TEMPNUM  
NUMBERT

CA  
INDEX  
MASK

TYPEP  
ROTEX  
JETSALL

TS  
MASK  
CCS

POLYTEMP  
CH6MASK  
A

TCF

+2

TC

Q

CCS

TEMPNUM

TCF

+4

TC

ALARM

OCT

02003

SELECTYZ

TCF  
TS  
TCF

JETSOFF  
NUMBERT  
SELECTP +1

# \*\*\*\*\* TCF ALARMJET \*\*\*\*\*

-1

JETSALL

TCF

ABORTYZ +2

OCT

00252

OCT

00125

# +P

OCT

00140

# -Y

OCT

00006

# -Z

OCT

00220

# +Y

OCT

00011

# +Z

OCT

00151

# +V

TYPEP

OCT

00146

# -U

OCT

00226

# -V

OCT

00231

# +U

OCT

00151

# +V

OCT

00132

# 1-3

OCT

00245

# 2-4

OCT

00377

# ALL

INDXYZ

-136MST6

=

-136MST6

DEC

-218

DEC

4

DEC

2

OCT

07776

DEC

5

DEC

9

DEC

10

OCT

07776

DEC

3

[illegible]

```
1
2 BANK 17
3 SETLOC DAPS2
4 BANK
5
6 EBANK= CDUXD
7
8 COUNT* $$/DAPQR
9
10 CALLQERR CA BIT13 # CALCULATE Q,R ERRORS UNLESS THESE AXES
11 EXTEND # ARE IN MANUAL RATE COMMAND.
12 RAND CHAN31
13
14 CCS A
15 TCF +5 # IN AUTO COMPUTE Q,R ERRORS
16 CS DAPBOOLS # IN MANUAL RATE COMMAND?
17
18 MASK OURRCBIT
19 EXTEND
20 BZF Q,RORGTS # IF SO BYPASS CALCULATION OF ERRORS.
21 TC QERRCALC
22
23 Q,RORGTS CCS COTROLER # CHOOSE CONTROL SYSTEM FOR THIS DAP PASS:
24 TCF GTOGTS # GTS (ALTERNATES WITH RCS WHEN DOCKED)
25 TCF TRYGTS # GTS IF ALLOWED, OTHERWISE RCS
26 RCS CAF ZERO # RCS (TRYGTS MAY BRANCH TO HERE)
27 TS COTROLER
28
29 DXCH EDOTQ
30 TC ROT-TOUV
31 DXCH OMEGAU
```

## # X - TRANSLATION

```
32 #
33 # INPUT: BITS 7,8 OF CH31 (TRANSLATION CONTROLLER)
34 #
35 # ULLAGER
36 # APSFLAG, DRIFTBIT
37 # ACC40R2X, ACRBTRAN
38
39 # OUTPUT: NEXTU, NEXTV CODES OF TRANSLATION FOR AFTER ROTATION
40 # SENSETYP TELL ROTATION DIRECTION AND DESIRE
41
42 # X-TRANS POLICIES ARE EITHER 4 JETS OR A DIAGONAL PAIR. IN 2-JET TRANSLATION THE SYSTEM IS SPECIFIED. A FAILURE
43 # WILL OVERRIDE THIS SPECIFICATION. AN ALARM RESULTS WHEN NO POLICY IS AVAILABLE BECAUSE OF FAILURES.
```

```
44 SENSEGET CA BIT7 # INPUT BITS OVERRIDE THE INTERNAL BITS
45 EXTEND # SENSETYP WILL NOT OPPOSE ANYTRANS
46 RAND CHAN31
47 EXTEND
48 BZF +XORULGE
```

|    |                          |        |          |                                        |    |
|----|--------------------------|--------|----------|----------------------------------------|----|
| 1  | # Q_R AXIS_RCS AUTOSTART |        |          | PAGE 1115                              | 1  |
| 2  |                          | CA     | BIT8     |                                        | 2  |
| 3  |                          | EXTEND |          |                                        | 3  |
| 4  |                          | RAND   | CHAN31   |                                        | 4  |
| 5  |                          | EXTEND |          |                                        | 5  |
| 6  |                          | BZF    | -XTRANS  |                                        | 6  |
| 7  |                          |        |          |                                        | 7  |
| 8  |                          | CA     | ULLAGER  |                                        | 8  |
| 9  |                          | MASK   | DAPBOOLS |                                        | 9  |
| 10 |                          | CCS    | A        |                                        | 10 |
| 11 |                          | TCF    | +XORULGE |                                        | 11 |
| 12 |                          |        |          |                                        | 12 |
| 13 |                          | TS     | NEXTU    | # STORE NULL TRANSLATION POLICIES      | 13 |
| 14 |                          | TS     | NEXTV    |                                        | 14 |
| 15 |                          | CS     | DAPBOOLS | # BURNING OR DRIFTING?                 | 15 |
| 16 |                          | MASK   | DRIFTBIT |                                        | 16 |
| 17 |                          | EXTEND |          |                                        | 17 |
| 18 |                          | BZF    | TSENSE   |                                        | 18 |
| 19 |                          | CA     | FLGWRD10 | # DPS (INCLUDING DOCKED) OR APS?       | 19 |
| 20 |                          | MASK   | APSFLBIT |                                        | 20 |
| 21 |                          | CCS    | A        |                                        | 21 |
| 22 |                          | CAF    | TWO      | # FAVOR +X JETS DURING AN APS BURN.    | 22 |
| 23 | TSENSE                   | TS     | SENSETYP |                                        | 23 |
| 24 |                          | TCF    | QRCONTRL |                                        | 24 |
| 25 |                          |        |          |                                        | 25 |
| 26 | +XORULGE                 | CAF    | ONE      |                                        | 26 |
| 27 | -XTRANS                  | AD     | FOUR     |                                        | 27 |
| 28 |                          | TS     | ROTINDEX |                                        | 28 |
| 29 |                          | AD     | NEG3     |                                        | 29 |
| 30 |                          | TS     | SENSETYP | # FAVOR APPROPRIATE JETS DURING TRANS. | 30 |
| 31 |                          | CA     | DAPBOOLS |                                        | 31 |
| 32 |                          | MASK   | ACC4OR2X |                                        | 32 |
| 33 |                          | CCS    | A        |                                        | 33 |
| 34 |                          | TCF    | TRANS4   |                                        | 34 |
| 35 |                          |        |          |                                        | 35 |
| 36 |                          | CA     | DAPBOOLS |                                        | 36 |
| 37 |                          | MASK   | AORBTRAN |                                        | 37 |
| 38 |                          | CCS    | A        |                                        | 38 |
| 39 |                          | CA     | ONE      | # THREE FOR B                          | 39 |
| 40 |                          | AD     | TWO      | # TWO FOR A SYSTEM 2 JET X TRANS       | 40 |
| 41 | TSENUMBRT                | TS     | NUMBERT  |                                        | 41 |
| 42 |                          |        |          |                                        | 42 |
| 43 |                          | TC     | SELCTSUB |                                        | 43 |
| 44 |                          |        |          |                                        | 44 |
| 45 |                          | CCS    | POLYTEMP |                                        | 45 |
| 46 |                          | TCF    | +3       |                                        | 46 |
| 47 |                          | TC     | ALARM    |                                        | 47 |
| 48 |                          | OCT    | 02002    |                                        | 48 |
| 49 |                          | CA     | 00314OCT |                                        | 49 |
| 50 |                          | MASK   | POLYTEMP |                                        | 50 |
| 51 | TSNEXTS                  | TS     | NEXTU    |                                        | 51 |
| 52 |                          |        |          |                                        | 52 |
| 53 |                          |        |          |                                        | 53 |
| 54 |                          |        |          |                                        | 54 |
| 55 |                          |        |          |                                        | 55 |
| 56 |                          |        |          |                                        | 56 |
| 57 |                          |        |          |                                        | 57 |
| 58 |                          |        |          |                                        | 58 |
| 59 |                          |        |          |                                        | 59 |
| 60 |                          |        |          |                                        | 60 |

```
1
2 CS 003140CT
3 MASK POLYTEMP
4 TS NEXTV
5
6 # Q,R-AXES RCS CONTROL MODE SELECTION
7 # SWITCHES INDICATION WHEN SET
8 # BIT13/CHAN31 AUTO, GO TO ATTSTEER
9 # PULSES MINIMUM IMPULSE MODE
10 # (OTHERWISE) RATE COMMAND/ATTITUDE HOLD MODE
11
12 QRCONTRL CA BIT13 # CHECK MODE SELECT SWITCH.
13
14 EXTEND
15 RAND CHAN31 # BITS INVERTED
16 CCS A
17
18 CHKBIT10 TCF ATTSTEER
19 CAF PULSES # PULSES = 1 FOR MIN IMP USE OF RHC
20 MASK DAPBOOLS
21
22 EXTEND
23 BZF CHEKSTIK # IN ATT-HOLD/RATE-COMMAND IF BIT10=0
24
25 # MINIMUM IMPULSE MODE
26
27 INHINT
28 TC IBNKCALL
29 CADR ZATTEROR
30 CA ZERO
31
32 TS QERROR
33 TS RERROR # FOR DISPLAYS
34 RELINT
35
36 EXTEND
37 READ CHAN31
38 TS TEMP31 # IS EQUAL TO DAPTEMP1
39 CCS OLDQRMIN
40 TCF CHECKIN
41
42 FIREQR CA TEMP31
43 MASK BIT1
44
45 EXTEND
46 BZF +QMIN
47
48 CA TEMP31
49 MASK BIT2
50 EXTEND
51 BZF -QMIN
52
53 CA TEMP31
54 MASK BIT5
```

|    |         |        |          |                                       |    |
|----|---------|--------|----------|---------------------------------------|----|
| 1  |         |        |          |                                       | 1  |
| 2  |         | EXTEND |          |                                       | 2  |
| 3  |         | BZF    | +RMIN    |                                       | 3  |
| 4  |         |        |          |                                       | 4  |
| 5  |         | CA     | TEMP31   |                                       | 5  |
| 6  |         | MASK   | BIT6     |                                       | 6  |
| 7  |         | EXTEND |          |                                       | 7  |
| 8  |         | BZF    | -RMIN    |                                       | 8  |
| 9  |         |        |          |                                       | 9  |
| 10 |         | TCF    | XTRANS   |                                       | 10 |
| 11 |         |        |          |                                       | 11 |
| 12 | CHECKIN | CS     | TEMP31   |                                       | 12 |
| 13 |         | MASK   | OCT63    |                                       | 13 |
| 14 |         | TS     | OLDQRMIN |                                       | 14 |
| 15 |         | TCF    | XTRANS   |                                       | 15 |
| 16 |         |        |          |                                       | 16 |
| 17 | +QMIN   | CA     | 14MS     |                                       | 17 |
| 18 |         | TS     | TJU      |                                       | 18 |
| 19 |         | CS     | 14MS     |                                       | 19 |
| 20 |         | TCF    | MINQR    |                                       | 20 |
| 21 | -QMIN   | CS     | 14MS     |                                       | 21 |
| 22 |         | TS     | TJU      |                                       | 22 |
| 23 |         | CA     | 14MS     |                                       | 23 |
| 24 |         | TCF    | MINQR    |                                       | 24 |
| 25 | +RMIN   | CA     | 14MS     |                                       | 25 |
| 26 |         | TCF    | +2       |                                       | 26 |
| 27 | -RMIN   | CS     | 14MS     |                                       | 27 |
| 28 |         | TS     | TJU      |                                       | 28 |
| 29 | MINQR   | TS     | TJV      |                                       | 29 |
| 30 |         | CA     | MINADR   |                                       | 30 |
| 31 |         | TS     | RETJADR  |                                       | 31 |
| 32 |         | CA     | ONE      |                                       | 32 |
| 33 |         | TS     | OLDQRMIN |                                       | 33 |
| 34 | MINRTN  | TS     | AXISCTR  |                                       | 34 |
| 35 |         | CA     | DAPBOOLS |                                       | 35 |
| 36 |         | MASK   | CSMDOCKD |                                       | 36 |
| 37 |         | EXTEND |          |                                       | 37 |
| 38 |         | BZF    | MIMRET   |                                       | 38 |
| 39 |         | INDEX  | AXISCTR  | # IF DOCKED, USE 60MS MINIMUM IMPULSE | 39 |
| 40 |         | CCS    | TJU      |                                       | 40 |
| 41 |         | CA     | 60MS     |                                       | 41 |
| 42 |         | TCF    | +2       |                                       | 42 |
| 43 |         | CS     | 60MS     |                                       | 43 |
| 44 |         | INDEX  | AXISCTR  |                                       | 44 |
| 45 |         | TS     | TJU      |                                       | 45 |
| 46 | MIMRET  | CA     | DAPBOOLS |                                       | 46 |
| 47 |         | MASK   | AORBTRAN |                                       | 47 |
| 48 |         | CCS    | A        |                                       | 48 |
| 49 |         | CA     | ONE      |                                       | 49 |
| 50 |         | AD     | TWO      |                                       | 50 |
| 51 |         | TS     | NUMBERT  |                                       | 51 |
| 52 |         |        |          |                                       | 52 |
| 53 |         |        |          |                                       | 53 |
| 54 |         |        |          |                                       | 54 |
| 55 |         |        |          |                                       | 55 |
| 56 |         |        |          |                                       | 56 |
| 57 |         |        |          |                                       | 57 |
| 58 |         |        |          |                                       | 58 |
| 59 |         |        |          |                                       | 59 |
| 60 |         |        |          |                                       | 60 |

```
1
2 TCF AFTERTJ
3
4 60MS DEC 96 # RSB 2009 -- WAS 96.0.
5 MINADR GENADR MINRTN
6 OCT63 OCT 63
7 14MS = +TJMINT6
8
9 TRANS4 CA FOUR
10 TCF TSNUMBRT
11
12 # RATE COMMAND MODE:
13 #
14 # DESCRIPTION (SAME AS P-AXIS)
15
16 CHEKSTIK TS INGT5 # NOT IN GTS WHEN IN ATT HOLD
17 CS ONE # 1/ACCS WILL DO THE NULLING DRIVES
18 TS COTROLER # COME BACK TO RCS NEXT TIME
19
20 CA BIT15
21 MASK CH31TEMP
22 EXTEND
23 BZF RHCACTIV # BRANCH IF OUT OF DETENT.
24 CA OURRCBIT # *****
25 MASK DAPBOOLS # *IN DETENT* CHECK FOR MANUAL CONTROL
26 EXTEND # ***** LAST TIME.
27 BZF STILLRCS
28 CS BIT9
29 MASK RCSFLAGS
30 TS RCSFLAGS # BIT 9 IS 0.
31 TCF DAMPING
32
33 40CYCL OCT 50
34 1/10S OCT 1
35 LINRAT DEC 46
36
37 # =====
38
39 DAMPING CA ZERO
40 TS SAVEHAND
41 TS SAVEHAND +1
42
43 RHCACTIV CCS SAVEHAND # *****
44 TCF +3 # Q,R MANUAL CONTROL WC = A*(B+|D|)*D
45 TCF +2 # *****
46 TCF +1
47 DOUBLE
48 DOUBLE # WHERE
49
50 AD LINRAT # WC = COMMANDED ROTATIONAL RATE
51 EXTEND # A = QUADRATIC SENSITIVITY FACTOR
52 MP SAVEHAND # B = LINEAR/QUADRATIC SENSITIVITY
53 CA L # |D| = ABS. VALUE OF DEFLECTION
54 EXTEND # D = HAND CONTROLLER DEFLECTION
55 MP STIKSENS
56 XCH QLAST # COMMAND Q RATE, SCALED 45 DEG/SEC
57 COM
58
59
60
```

```
1
2 AD QLAST
3 TS DAPTEMP3
4 CCS SAVEHAND +1
5 TCF +3
6 TCF +2
7 TCF +1
8 DOUBLE
9 DOUBLE
10 AD LINRAT
11 EXTEND
12 MP SAVEHAND +1
13 CA L
14 EXTEND
15 MP STIKSENS
16 XCH RLAST
17 COM
18 AD RLAST
19 TS DAPTEMP4
20 CS QLAST # INTERVAL.
21 AD OMEGAQ
22 TS QRATEDIF
23 CS RLAST
24 AD OMEGAR
25 TS RRATEDIF
26 DXCH QRATEDIF # TRANSFORM RATES FROM Q,R TO U,V AXES
27 TC ROT-TOUV
28 DXCH URATEDIF
29 CCS DAPTEMP3 # CHECK IF Q COMMAND CHANGE EXCEEDS
30 TC +3 # BREAKOUT LEVEL. IF NOT, CHECK R.
31 TC +2
32 TC +1
33 AD -RATEDB
34 EXTEND
35 BZMF +2
36 TCF ENTERUV -2 # BREAKOUT LEVEL EXCEEDED. DIRECT RATE.
37 CCS DAPTEMP4 # R COMMAND BREAKOUT CHECK.
38 TC +3
39 TC +2
40 TC +1
41 AD -RATEDB
42 EXTEND
43 BZMF +2
44 TCF ENTERUV -2 # BREAKOUT LEVEL EXCEEDED. DIRECT RATE.
45 CA RCSFLAGS # BREAKOUT LEVEL NOT EXCEEDED. CHECK FOR
46 MASK QRBIT # DIRECT RATE CONTROL LAST TIME.
47 EXTEND
48 BZF +2
49 TCF ENTERUV # CONTINUE DIRECT RATE CONTROL.
50 TCF STILLRCS # PSEUDO-AUTO CONTROL.
51 CA 40CYCL
52
53
54
55
56
57
58
59
60
```



|    |          |        |            |                                         |
|----|----------|--------|------------|-----------------------------------------|
| 1  |          |        |            |                                         |
| 2  |          | TS     | TCQR       |                                         |
| 3  | ENTERUV  | INHINT |            | # DIRECT RATE CONTROL                   |
| 4  |          | TC     | IBNKCALL   |                                         |
| 5  |          | FCADR  | ZATTEROR   |                                         |
| 6  |          | RELINT |            |                                         |
| 7  |          | CA     | ZERO       |                                         |
| 8  |          | TS     | DYERROR    |                                         |
| 9  |          | TS     | DYERROR +1 |                                         |
| 10 |          | TS     | DZERROR    |                                         |
| 11 |          | TS     | DZERROR +1 |                                         |
| 12 |          | CCS    | URATEDIF   |                                         |
| 13 |          | TCF    | +3         |                                         |
| 14 |          | TCF    | +2         |                                         |
| 15 |          | TCF    | +1         |                                         |
| 16 |          | AD     | TARGETDB   | # IF TARGET DB IS EXCEEDED, CONTINUE    |
| 17 |          | EXTEND |            | # DIRECT RATE CONTROL.                  |
| 18 |          | BZMF   | VDB        |                                         |
| 19 |          | CCS    | VRATEDIF   |                                         |
| 20 |          | TCF    | +3         |                                         |
| 21 |          | TCF    | +2         |                                         |
| 22 |          | TCF    | +1         |                                         |
| 23 |          | AD     | TARGETDB   |                                         |
| 24 |          | EXTEND |            |                                         |
| 25 |          | BZMF   | +2         |                                         |
| 26 |          | TCF    | QRTIME     |                                         |
| 27 |          | CA     | ZERO       |                                         |
| 28 |          | TS     | VRATEDIF   |                                         |
| 29 |          | TCF    | QRTIME     |                                         |
| 30 | VDB      | CCS    | VRATEDIF   |                                         |
| 31 |          | TC     | +3         |                                         |
| 32 |          | TC     | +2         |                                         |
| 33 |          | TC     | +1         |                                         |
| 34 |          | AD     | TARGETDB   | # IF TARGET DB IS EXCEEDED, CONTINUE    |
| 35 |          | EXTEND |            | # DIRECT RATE CONTROL. IF NOT, FIRE AND |
| 36 |          | BZMF   | TOPSEUDO   | # SWITCH TO PSEUDO-AUTO CONTROL ON NEXT |
| 37 |          | CA     | ZERO       | # PASS.                                 |
| 38 |          | TS     | URATEDIF   |                                         |
| 39 | QRTIME   | CA     | TCQR       | # DIRECT RATE TIME CHECK.               |
| 40 |          | EXTEND |            |                                         |
| 41 |          | BZMF   | +5         | # BRANCH IF TIME EXCEEDS 4 SEC.         |
| 42 |          | CS     | RCSFLAGS   |                                         |
| 43 |          | MASK   | QRBIT      |                                         |
| 44 |          | ADS    | RCSFLAGS   | # BIT 11 IS 1.                          |
| 45 |          | TC     | +4         |                                         |
| 46 | TOPSEUDO | CS     | QRBIT      |                                         |
| 47 |          | MASK   | RCSFLAGS   |                                         |
| 48 |          | TS     | RCSFLAGS   | # BIT 11 IS 0.                          |
| 49 |          | CA     | HANDADR    |                                         |
| 50 |          | TS     | RETJADR    |                                         |
| 51 |          | CA     | ONE        |                                         |
| 52 |          |        |            |                                         |
| 53 |          |        |            |                                         |
| 54 |          |        |            |                                         |
| 55 |          |        |            |                                         |
| 56 |          |        |            |                                         |
| 57 |          |        |            |                                         |
| 58 |          |        |            |                                         |
| 59 |          |        |            |                                         |
| 60 |          |        |            |                                         |

|                              |        |            |   |                                           |      |               |
|------------------------------|--------|------------|---|-------------------------------------------|------|---------------|
| BACKHAND                     | TS     | AXISCTR    |   |                                           |      |               |
|                              | CA     | FOUR       |   |                                           |      |               |
|                              | TS     | NUMBERT    |   |                                           |      |               |
|                              | INDEX  | AXISCTR    |   |                                           |      |               |
|                              | INDEX  | SKIPU      |   |                                           |      |               |
|                              | TCF    | +1         |   |                                           |      |               |
|                              | CA     | FOUR       |   |                                           |      |               |
|                              | INDEX  | AXISCTR    |   |                                           |      |               |
|                              | TS     | SKIPU      |   |                                           |      |               |
|                              | TCF    | LOOPER     |   |                                           |      |               |
|                              | INDEX  | AXISCTR    |   |                                           |      |               |
|                              | CCS    | URATEDIF   | # | INDEX                                     | AXIS | QUANTITY      |
|                              | CA     | ZERO       | # | 0                                         | -U   | 1/JETACC-AOSU |
|                              | TCF    | +2         | # | 1                                         | +U   | 1/JETACC+AOSU |
|                              | CA     | ONE        | # | 16                                        | -V   | 1/JETACC-AOSV |
|                              | INDEX  | AXISCTR    | # | 17                                        | +V   | 1/JETACC+AOSV |
|                              | AD     | AXISDIFF   | # | JETACC = 2 JET ACCELERATION (1 FOR FAIL)  |      |               |
|                              | INDEX  | A          |   |                                           |      |               |
|                              | CS     | 1/ANET2 +1 |   |                                           |      |               |
|                              | EXTEND |            |   |                                           |      |               |
|                              | INDEX  | AXISCTR    | # | UPRATEDIF IS SCALED AT PI/4 RAD/SEC       |      |               |
|                              | MP     | URATEDIF   | # | JET TIME IN A, SCALED 32 SEC              |      |               |
|                              | TS     | Q          |   |                                           |      |               |
|                              | DAS    | A          |   |                                           |      |               |
|                              | AD     | Q          |   |                                           |      |               |
|                              | TS     | A          | # | OVERFLOW SKIP                             |      |               |
|                              | TCF    | +2         |   |                                           |      |               |
|                              | CA     | Q          | # | RIGHT SIGN AND BIGGER THAN 150MS          |      |               |
| SETTIME                      | INDEX  | AXISCTR    |   |                                           |      |               |
|                              | TS     | TJU        | # | SCALED AT 10.67 WHICH IS CLOSE TO 10.24   |      |               |
|                              | TCF    | AFTERTJ    |   |                                           |      |               |
| ZEROTJ                       | CA     | ZERO       |   |                                           |      |               |
|                              | TCF    | SETTIME    |   |                                           |      |               |
| HANDADR                      | GENADR | BACKHAND   |   |                                           |      |               |
| # GTS WILL BE TRIED IF       |        |            |   |                                           |      |               |
| # 1. USEQRJTS = 0,           |        |            |   |                                           |      |               |
| # 2. ALLOWGTS POS,           |        |            |   |                                           |      |               |
| # 3. JETS ARE OFF (Q,R-AXES) |        |            |   |                                           |      |               |
| TRYGTS                       | CAF    | USEQRJTS   | # | IS JET USE MANDATORY. (AS LONG AS         |      |               |
|                              | MASK   | DAPBOOLS   | # | USEQRJTS BIT IS NOT BIT 15, CCS IS SAFE.) |      |               |
|                              | CCS    | A          |   |                                           |      |               |
|                              | TCF    | RCS        |   |                                           |      |               |
|                              | CCS    | ALLOWGTS   | # | NO. DOES AOSTASK OK CONTROL FOR GTS?      |      |               |

[illegible]

# SUBROUTINE TO COMPUTE Q,R-AXES ATTITUDE ERRORS FOR USE IN THE RCS AND GTS CONTROL LAWS AND THE DISPLAYS.

|          |        |           |                                         |
|----------|--------|-----------|-----------------------------------------|
| QERRCALC | CAE    | CDUY      | # Q-ERROR CALCULATION                   |
|          | EXTEND |           |                                         |
|          | MSU    | CDUYD     | # CDU ANGLE -- ANGLE DESIRED (Y-AXIS)   |
|          | TS     | DAPTEMP1  | # SAVE FOR RERRCALC                     |
|          | EXTEND |           |                                         |
|          | MP     | M21       | # (CDUY-CDUYD)*M21 SCALED AT PI RADIANS |
|          | TS     | E         |                                         |
|          | CAE    | CDUZ      | # SECOND TERM CALCULATION:              |
|          | EXTEND |           |                                         |
|          | MSU    | CDUZD     | # CDU ANGLE -ANGLE DESIRED (Z-AXIS)     |
|          | TS     | DAPTEMP2  | # SAVE FOR RERRCALC                     |
|          | EXTEND |           |                                         |
|          | MP     | M22       | # (CDUZ-CDUZD)*M22 SCALED AT PI RADIANS |
|          | AD     | DELQEROR  | # KALCMANU INERFACE ERROR               |
|          | AD     | E         |                                         |
|          | XCH    | QERROR    | # SAVE Q-ERROR FOR EIGHT-BALL DISPLAY.  |
| RERRCALC | CAE    | DAPTEMP1  | # R-ERROR CALCULATION:                  |
|          | EXTEND |           | # CDU ANGLE -ANGLE DESIRED (Y-AXIS)     |
|          | MP     | M31       | # (CDUY-CDUYD)*M31 SCALED AT PI RADIANS |
|          | TS     | E         |                                         |
|          | CAE    | DAPTEMP2  | # SECOND TERM CALCULATION:              |
|          | EXTEND |           | # CDU ANGLE -ANGLE DESIRED (Z-AXIS)     |
|          | MP     | M32       | # (CDUZ-CDUZD)*M32 SCALED AT PI RADIANS |
|          | AD     | DELRREROR | # KALCMANU INERFACE ERROR               |
|          | AD     | E         |                                         |
|          | XCH    | RERROR    | # SAVE R-ERROR FOR EIGHT-BALL DISPLAY.  |
|          | TC     | Q         |                                         |

# "ATTSTEER" IS THE ENTRY POINT FOR Q,R-AXES (U,V-AXES) ATTITUDE CONTROL USING THE REACTION CONTROL SYSTEM

ATTSTEER        EQUALS    STILLRCS        # "STILLRCS" IS THE RCS EXIT FROM TRYGTS.

STILLRCS        CA        RERROR  
                 LXCH     A  
                 CA        QERROR  
                 TC        ROT-TOUV  
                 DXCH     UERROR

# PREPARES CALL TO TJETLAW (OR SPSRCS(DOCKED))

# PREFORMS SKIP LOGIC ON U OR Y AXIS IF NEEDED.

TJLAW            CA        TJLAWADR  
                 TS        RETJADR  
                 CA        ONE  
                 TS        AXISCTR  
                 INDEX    AXISCTR  
                 INDEX    SKIPU  
                 TCF        +1

                 CA        FOUR  
                 INDEX    AXISCTR  
                 TS        SKIPU  
                 TCF        LOOPER  
                 INDEX    AXISCTR  
                 CA        UERROR

                 TS        E  
                 INDEX    AXISCTR  
                 CA        OMEGAU  
                 TS        EDOT  
                 CA        DAPBOOLS  
                 MASK     CSMDOCKD

                 CCS        A  
                 TCF        +3  
                 TC        TJETLAW

+3                TCF        AFTERTJ  
                 CS        DAPBOOLS        # DOCKED. IF GIMBAL USABLE DO GTS CONTROL  
                 MASK     USEQRJTS        # ON THE NEXT PASS.

                 CCS        A        # USEQRJTS BIT MUST NOT BE BIT 15.  
                 TS        COTROLER    # GIMBAL USABLE. STORE POSITIVE VALUE.  
                 INHINT

                 TC        IBNKCALL  
                 CADR     SPSRCS        # DETERMINE RCS CONTROL  
                 RELINT

                 CAF        FOUR        # ALWAYS CALL FOR 2-JET CONTROL ABOUT U,V.  
                 TS        NUMBERT    # FALL THROUGH TO JET SELECTION, ETC.

# Q,R-JET-SELECTION-LOGIC

#

# INPUT:        AXISCTR        0,1 FOR U,V

#                SNUFFBIT        ZERO TJETU,V AND TRANS. ONLY IF SET IN A DPS BURN

```
1 #
2 # TJU,TJV JET TIME SCALED 10.24 SEC.
3 # NUMBERT INDICATES NUMBER OF JETS AND TYPE OF POLICY
4 # RETJADR WHERE TO RETURN TO
5 #
6 # OUTPUT: NO.U(V)JETS RATE DERIVATION FEEDBACK
7 # CHANNEL 5
8 # SKIPU,SKIPV FOR LESS THAN 150MS FIRING
9 #
10 # NOTES: IN CASE OF FAILURE IN DESIRED ROTATION POLICY, "ALL" UNFAILED
11 # JETS OF THE DESIRED POLICY ARE SELECTED. SINCE THERE ARE ONLY
12 # TWO JETS, THIS MEANS THE OTHER ONE OR NONE. THE ALARM IS SENT
13 # IF NONE CAN BE FOUND.
14 #
15 # TIMES LESS THAN 14 MSEC ARE TAKEN TO CALL FOR A SINGLE-JET
16 # MINIMUM IMPULSE, WITH THE JET CHOSEN SEMI-RANDOMLY.
17
18 AFTERTJ CA FLAGWRD5 # IF SNUFFBIT SET DURING A DPS BURN GO TO
19 MASK SNUFFBIT # XTRANS; THAT IS, INHIBIT CONTROL.
20 EXTEND
21 BZF DOROTAT
22 CS FLGWRD10
23 MASK APSFLBIT
24 EXTEND
25 BZF DOROTAT
26 CA DAPBOOLS
27 MASK DRIFTBIT
28 EXTEND
29 BZF XTRANS
30
31 DOROTAT CAF TWO
32 TS L
33 INDEX AXISCTR
34 CCS TJU
35 TCF +5
36 TCF NOROTAT
37 TCF +2
38 TCF NOROTAT
39 ZL
40 AD ONE
41 TS ABSTJ
42
43 CA AXISCTR
44 AD L
45 TS ROTINDEX # 0 1 2 3 = -U -V +U +V
46
47 CA ABSTJ
48 AD -150MS
49 EXTEND
50 BZMF DOSKIP
51
52
53
54
55
56
57
58
59
60
```

|                           |         |           |                                                 |
|---------------------------|---------|-----------|-------------------------------------------------|
| # Q-R AXIS-ROTOR-ROTATION |         | PAGE 1121 |                                                 |
| 2                         | TC      | SELCTSUB  |                                                 |
| 3                         |         |           |                                                 |
| 4                         | INDEX   | AXISCTR   |                                                 |
| 5                         | CA      | INDEXES   |                                                 |
| 6                         | TS      | L         |                                                 |
| 7                         |         |           |                                                 |
| 8                         | CA      | POLYTEMP  |                                                 |
| 9                         | INHINT  |           |                                                 |
| 10                        | INDEX   | L         |                                                 |
| 11                        | TC      | WRITEP    |                                                 |
| 12                        |         |           |                                                 |
| 13                        | RELINT  |           |                                                 |
| 14                        | TCF     | FEEDBACK  |                                                 |
| 15                        |         |           |                                                 |
| 16                        | NOROTAT | INDEX     | AXISCTR                                         |
| 17                        |         | CA        | INDEXES                                         |
| 18                        |         | INHINT    |                                                 |
| 19                        |         | INDEX     | A                                               |
| 20                        |         | TC        | WRITEP -1                                       |
| 21                        |         |           |                                                 |
| 22                        |         | RELINT    |                                                 |
| 23                        | LOOPER  | CCS       | AXISCTR                                         |
| 24                        |         | TC        | RETJADR                                         |
| 25                        |         | TCF       | CLOSEOUT                                        |
| 26                        | DOSKIP  | CS        | ABSTJ                                           |
| 27                        |         | AD        | +TJMINT6 # 14MS                                 |
| 28                        |         | EXTEND    |                                                 |
| 29                        |         | BZMF      | NOTMIN                                          |
| 30                        |         |           |                                                 |
| 31                        |         | ADS       | ABSTJ                                           |
| 32                        |         | INDEX     | AXISCTR                                         |
| 33                        |         | CCS       | TJU                                             |
| 34                        |         | CA        | +TJMINT6                                        |
| 35                        |         | TCF       | +2                                              |
| 36                        |         | CS        | +TJMINT6                                        |
| 37                        |         | INDEX     | AXISCTR                                         |
| 38                        |         | TS        | TJU                                             |
| 39                        |         |           |                                                 |
| 40                        |         | CCS       | SENSETYP # ENSURE MIN-IMPULSE NOT AGAINST TRANS |
| 41                        |         | TCF       | NOTMIN -1                                       |
| 42                        |         | EXTEND    |                                                 |
| 43                        |         | READ      | LOSCALAR                                        |
| 44                        |         | MASK      | ONE                                             |
| 45                        |         | TS        | NUMBERT                                         |
| 46                        |         |           |                                                 |
| 47                        | NOTMIN  | TC        | SELCTSUB                                        |
| 48                        |         |           |                                                 |
| 49                        |         | INDEX     | AXISCTR                                         |
| 50                        |         | CA        | INDEXES                                         |
| 51                        |         | INHINT    |                                                 |
| 52                        |         |           |                                                 |
| 53                        |         |           |                                                 |
| 54                        |         |           |                                                 |
| 55                        |         |           |                                                 |
| 56                        |         |           |                                                 |
| 57                        |         |           |                                                 |
| 58                        |         |           |                                                 |
| 59                        |         |           |                                                 |
| 60                        |         |           |                                                 |

|    |                       |        |                           |    |
|----|-----------------------|--------|---------------------------|----|
| 1  | # QR AXIS ROO ACUTEST |        |                           | 1  |
| 2  |                       | TS     | T6FURTHA +1               | 2  |
| 3  |                       | CA     | POLYTEMP                  | 3  |
| 4  |                       | INDEX  | T6FURTHA +1               | 5  |
| 5  |                       | TC     | WRITEP                    | 6  |
| 6  |                       |        |                           | 7  |
| 7  |                       | CA     | ABSTJ                     | 9  |
| 8  |                       | TS     | T6FURTHA                  | 10 |
| 9  |                       | TC     | JTLST # IN QR BANK BY NOW | 12 |
| 10 |                       |        |                           | 13 |
| 11 |                       | RELINT |                           | 14 |
| 12 |                       |        |                           | 15 |
| 13 |                       | CA     | ZERO                      | 17 |
| 14 |                       | INDEX  | AXISCTR                   | 18 |
| 15 |                       | TS     | SKIPU                     | 19 |
| 16 |                       |        |                           | 20 |
| 17 | FEEDBACK              | CS     | THREE                     | 22 |
| 18 |                       | AD     | NUMBERT                   | 24 |
| 19 |                       | EXTEND |                           | 25 |
| 20 |                       | BZMF   | +3                        | 26 |
| 21 |                       |        |                           | 28 |
| 22 |                       | CA     | TWO                       | 29 |
| 23 |                       | TCF    | +2                        | 30 |
| 24 |                       | CA     | ONE                       | 32 |
| 25 |                       | INDEX  | AXISCTR                   | 33 |
| 26 |                       | TS     | NO.UJETS                  | 34 |
| 27 |                       | TCF    | LOOPER                    | 36 |
| 28 |                       |        |                           | 37 |
| 29 | XTRANS                | CA     | ZERO                      | 38 |
| 30 |                       | TS     | TJU                       | 39 |
| 31 |                       | TS     | TJV                       | 41 |
| 32 |                       | CA     | FOUR                      | 42 |
| 33 |                       | INHINT |                           | 44 |
| 34 |                       | XCH    | SKIPU                     | 45 |
| 35 |                       | EXTEND |                           | 46 |
| 36 |                       | BZF    | +2                        | 47 |
| 37 |                       | TC     | WRITEU -1                 | 49 |
| 38 |                       | CA     | FOUR                      | 50 |
| 39 |                       | XCH    | SKIPV                     | 52 |
| 40 |                       | RELINT |                           | 53 |
| 41 |                       |        |                           | 54 |
| 42 |                       | EXTEND |                           | 55 |
| 43 |                       | BZF    | CLOSEOUT                  | 57 |
| 44 |                       | INHINT |                           | 58 |
| 45 |                       | TC     | WRITEV -1                 | 59 |
| 46 |                       | RELINT |                           | 61 |
| 47 |                       |        |                           | 62 |
| 48 |                       | TCF    | CLOSEOUT                  | 64 |
| 49 | INDEXES               | DEC    | 4                         | 65 |
| 50 |                       | DEC    | 13                        | 66 |
| 51 | +TJMINT6              | DEC    | 22                        | 68 |
| 52 |                       |        |                           | 69 |
| 53 |                       |        |                           | 70 |
| 54 |                       |        |                           | 72 |
| 55 |                       |        |                           | 73 |
| 56 |                       |        |                           | 74 |
| 57 |                       |        |                           | 76 |
| 58 |                       |        |                           | 77 |
| 59 |                       |        |                           | 78 |
| 60 |                       |        |                           | 80 |



```
1 -150MS DEC -240
2 BIT8,9 OCT 00600
3
4 SCLNORM OCT 266
5 TJLAWADR GENADR TJLAW +3 # RETURN ADDRESS FOR RCS ATTITUDE CONTROL
6
```

```
7 # THE JET LIST:
8 # THIS IS A WAITLIST FOR T6RUPTS.
9 #
```

```
10 # CALLED BY:
11 # CA TJ # TIME WHEN NEXT JETS WILL BE WRITTEN
12 # TS T6FURTHA
13 # CA INDEX # AXIS TO BE WRITTEN AT TJ (FROM NOW)
14 # TS T6FURTHA +1
15 # TC JTLST
```

```
16 #
17 # EXAMPLE -- U-AXIS AUTOPILOT WILL WRITE ITS ROTATION CODE OF
18 # JETS INTO CHANNEL 5. IF IT DESIRES TO TURN OFF THIS POLICY WITHIN
19 # 150MS AND THEN FIRE NEXTU, A CALL TO JTLST IS MADE WITH T6FURTHA
20 # CONTAINING THE TIME TO TURN OFF THE POLICY, T6FURTHA +1 THE INDEX
21 # OF THE U-AXIS(4), AND NEXTU WILL CONTAIN THE "U-TRANS" POLICY OR ZERO.
```

```
22 #
23 # THE LIST IS EXACTLY 3 LONG. (THIS LEADS UP TO SKIP LOGIC AND 150MS LIMIT)
24 # THE INPUT IS THE LAST MEMBER OF THE LIST.
```

```
25 #
26 # RETURNS BY:
27 # + TC Q
```

```
28 #
29 # DEFINITIONS: (OUTPUT)
30 # TIME6 TIME OF NEXT RUPT
31 # T6NEXT DELTA TIME TO NEXT RUPT
32 # T6FURTHA DELTA TIME FROM 2ND TO LAST RUPT
33 # NXT6ADR AXIS INDEX 0 -- P-AXIS
34 # T6NEXT +1 AXIS INDEX 4 -- U-AXIS
35 # T6FURTHA +1 AXIS INDEX 13 -- V-AXIS
```

```
36
37 JTLST CS T6FURTHA
38 AD TIME6
39 EXTEND
40 BZMF MIDORLST # TIME6 -- TI IS IN A
```

```
41
42 LXCH NXT6ADR
43 DXCH T6NEXT
44 DXCH T6FURTHA
45 TS TIME6
46 LXCH NXT6ADR
```

```
47
48 TURNON CA BIT15
49 EXTEND
50 WOR CHAN13
51 TC Q
```

```
1 MIDORLST AD T6NEXT
2 AD T6NEXT
3 EXTEND
4 BZMF LASTCHG # TIME6 + T6NEXT - T IS IN A
5
6 LXCH T6NEXT +1
7 DXCH T6FURTHA
8 EXTEND
9 SU TIME6
10 DXCH T6NEXT
11
12 TC Q
13
14 LASTCHG CS A
15 AD NEG0
16 TS T6FURTHA
17
18 TC Q
19
20 # ROT-TOUV IS ENTERED WITH THE Q-COMPONENT OF THE QUANTITY TO BE TRANSFORMED IN A AND THE R-COMPONENT IN L.
21 # ROT-TOUV TRANSFORMS THE QUANTITY INTO THE NON-ORTHOGONAL U-V AXIS SYSTEM. IN THE U-V SYSTEM NO CROSS-COUPPLING IS
22 # PRODUCED FROM RCS JET FIRINGS. AT THE COMPLETION OF ROT-TOUV, THE U-COMPONENT OF THE TRANSFORMED QUANTITY IS IN
23 # A AND THE V-COMPONENT IS IN L.
24
25 ROT-TOUV LXCH ROTEMP2 # (R) IS PUT INTO ROTEMP2
26 EXTEND
27 MP COEFFQ
28 XCH ROTEMP2 # (R) GOES TO A AND COEFFQ.(Q) TO ROTEMP2
29 EXTEND
30 MP COEFFR
31 TS L # COEFFR.(R) IS PUT INTO L
32 AD ROTEMP2
33 TS ROTEMP1 # COEFFQ.(Q)+COEFFR.(R) IS PUT IN ROTEMP1
34 TCF +4
35 INDEX A # COEFFQ.(Q) + COEFFR.(R) HAS OVERFLOWED
36 CS LIMITS # AND IS LIMITED TO POSMAX OR NEGMAX
37 TS ROTEMP1
38 CS ROTEMP2
39 AD L # -COEFFQ.(Q) + COEFFR.(R) IS NOW IN A
40 TS 7
41 TCF +3
42 INDEX A # -COEFFQ.(Q) + COEFFR.(R) HAS OVERFLOWED
43 CS LIMITS # AND IS LIMITED TO POSMAX OR NEGMAX
44 LXCH ROTEMP1 # COEFFQ.(Q) + COEFFR.(R) IS PUT INTO L
45 TC Q
46
47 SELCTSUB INDEX ROTINDEX
48 CA ALLJETS
49 INDEX NUMBERT
50 MASK TYPEPOLY
51 TS POLYTEMP
```

[illegible]

|        |        |
|--------|--------|
| BLOCK  | 3      |
| SETLOC | FFTAG6 |
| BANK   |        |

COUNT\* \$\$/DAP

|          |        |          |
|----------|--------|----------|
| MAKERUPT | EXTEND |          |
|          | EDRUPT | MAKERUPT |

## # PROGRAM DESCRIPTION

# DESIGNED BY: R. D. GOSS AND P. S. WEISSMAN

# CODED BY: P. S. WEISSMAN, 28 FEBRUARY 1968

#

# TJETLAW IS CALLED AS A SUBROUTINE WHEN THE LEM IS NOT DOCKED AND THE AUTOPILOT IS IN THE AUTOMATIC OR

# ATTITUDE-HOLD MODE TO CALCULATE THE JET-FIRING-TIME (TJET) REQUIRED FOR THE AXIS INDICATED BY AXISCTR:

# -1 INDICATES THE P-AXIS

# +0 INDICATES THE U-AXIS

# +1 INDICATES THE V-AXIS

# THE REGISTERS E AND EDOT CONTAIN THE APPROPRIATE ATTITUDE ERROR AND ERROR RATE AND SENSETYP SHOWS WHETHER

# UNBALANCED COUPLES ARE PREFERRED. TJETLAW ALSO USES VARIOUS FUNCTIONS OF ACCELERATION AND DEADBAND WHICH ARE

# COMPUTED IN THE 1/ACCONT SECTION OF 1/ACCS AND ARE STORED IN SUCH AN ORDER THAT THEY CAN BE CONVENIENTLY

# ACCESSED BY INDEXING.

#

# THE SIGN OF THE REQUIRED ROTATION IS CARRIED THROUGH TJETLAW AS ROTSENSE AND IS FINALLY APPLIED TO TJET JUST

# PREVIOUS TO ITS STORAGE IN THE LOCATION CORRESPONDING TO THE AXIS (TJP, TJU, OR TJV). THE NUMBER OF JETS THAT

# TJETLAW ASSUMES WILL BE USED AS INDICATED BY THE SETTING OF NUMBERT FOR THE U- OR V-AXIS. TWO JETS ARE ALWAYS

# ASSUMED FOR THE P-AXIS ALTHOUGH FOUR JETS WILL BE FIRED WHEN FIREFCT IS MORE NEGATIVE THAN -4.0 DEGREES

# (FIREFCT IS THE DISTANCE TO A SWITCH CURVE IN THE PHASE PLANE) AND A LONG FIRING IS CALLED FOR.

#

# IN ORDER TO AVOID SCALING DIFFICULTIES, SIMPLE ALGORITHMS TAGGED RUFLAW1, -2 AND -3 ARE RESORTED TO WHEN THE

# ERROR AND/OR ERROR RATE ARE LARGE.

#

## # CALLING SEQUENCE:

# TC TJETLAW # (MUST BE IN JASK)

# OR

# INHINT # (MUST BE IN JASK)

# TC IBNKCALL

# CADR TJETLAW

# RELINT

#

# EXIT: RETURN TO Q.

#

## # INPUT:

# FROM THE CALLER: E, EDOT, AXISCTR, SENSETYP, TJP, -U, -V.

# FROM 1/ACCONT: 48 ERASABLES BEGINNING AT BLOCKTOP (INCLUDING FLAT, ZONE3LIM AND ACCSWU, -V).

#

## # OUTPUT:

# TJP, -U OR -V, NUMBERT (DAPTEMP5), FIREFCT (DAPTEMP3).

#

## # DEBRIS:

# A, L, Q, E, EDOT, DAPTEMP1-6, DAPTEMP1-4.

#

# ALARM: NONE

BANK 17

SETLOC DAPS2

BANK

EBANK= TJP

```
1
2 COUNT* $$/DAPTJ
3
4 TJETLAW EXTEND # SAVE Q FOR RETURN.
5 QXCH HOLDQ
6
7 # SET INDEXERS TO CORRESPOND TO THE AXIS AND TO THE SIGN OF EDOT
8
9 INDEX AXISCTR # AXISDIFF(-1)=NO OF LOCATIONS BET P AND U
10 CAF AXISDIFF # AXISDIFF(0)=0
11 TS ADRSDIF1 # AXISDIFF(+1)=NO OF LOCATIONS BET V AND U
12
13 CAE EDOT # IF EDOT NEGATIVE, PICK UP SET OF VALUES
14 EXTEND # THAT ALLOW USE OF SAME CODING AS FOR
15 BZMF NEGEDOT # POSITIVE EDOT.
16 CAE ADRSDIF1 # SET A SECOND INDEXER WHICH MAY BE
17 TS ADRSDIF2 # MODIFIED BY A DECISION FOR MAX JETS.
18 CAF SENSOR # FOR POSITIVE EDOT, ROTSENSE IS
19 TCF SETSENSE # INITIALIZED POSITIVE.
20
21 NEGEDOT CS E # IN ORDER FOR NEG EDOT CASE TO USE CODING
22 TS E # OF POS EDOT, MUST MODIFY AS FOLLOWS:
23 CS EDOT # 1. COMPLEMENT E AND EDOT.
24 TS EDOT # 2. SET SENSE OF ROTATION TO NEGATIVE
25 CAF BIT1 # (REVERSED LATER IF NECESSARY).
26 ADS ADRSDIF1 # 3. INCREMENT INDEXERS BY ONE SO THAT
27 TS ADRSDIF2 # THE PROPER PARAMETERS ARE ACCESSED.
28
29 SETSENSE CS SENSOR
30 TS ROTSENSE
31
32 # TEST MAGNITUDE OF E (ATTITUDE ERROR, SINGLE-PRECISION, SCALED AT PI RADIANS):
33 # IF GREATER THAN (OR EQUAL TO) PI/16 RADIANS, GO TO THE SIMPLIFIED TJET ROUTINE.
34 # IF LESS THAN PI/16 RADIANS, RESCALE TO PI/4
35
36 CAE E # PICK UP ATTITUDE ERROR FOR THIS AXIS
37 EXTEND
38 MP BIT5 # SHIFT RIGHT TEN BITS: IF A-REGISTER IS
39 CCS A # ZERO, RESCALE AND TEST EDOT.
40 TCF RUFLAW2
41
42 SCALEE TCF SCALEE
43 TCF RUFLAW1
44 CAF BIT13 # ERROR IS IN L SCALED AT PI/16. RESCALE
45 EXTEND # IT TO PI/4 AND SAVE IT.
46 MP L
47 TS E
48
49 # TEST MAGNITUDE OF EDOT (ERROR RATE SCALED AT PI/4 RADIANS/SECOND)
50 # IF GREATER THAN (OR EQUAL TO) PI/32 RADIANS/SECOND, GO TO THE SIMPLIFIED TJET ROUTINE.
51 # IF LESS THAN PI/32 RADIANS/SECOND, THEN RESCALE TO PI/32 RADIANS/SECOND.
52
53 CAE EDOT # PICK UP SINGLE-PRECISION ERROR-RATE
```

```
1
2 EXTEND # FOR THIS AXIS=
3 MP # SHIFT RIGHT ELEVEN BITS, IF THE A-REG IS
4 BIT4 # ZERO, THEN RESCALE AND USE FINELAW.
5 EXTEND
6 BZF SCALEDOT
7 TCF RUFLAW3
8
9 # *** FINELAW STARTS HERE ***
10
11 SCALEDOT LXCH EDOT # EDOT IS SCALED AT PI/32 RADIANS/SECOND.
12
13 CAE EDOT # COMPUTE (EDOT)(EDOT)
14 EXTEND
15 SQUARE
16 EXTEND # PRODUCT SCALED AT PI(2)/2(10) RAD/SEC.
17
18 MP BIT13
19 TS EDOTSQ # SHIFT RIGHT TWO BITS TO RESCALE TO EDOTSQ
20 # TO PI(2)/2(8) RAD(2)/SEC(2).
21
22 ERRTEST CCS E # DOES BIG ERROR (THREE DEG BEYOND THE
23 AD -3DEG # DEADBAND) REQUIRE MAXIMUM JETS?
24 TCF +2
25 AD -3DEG
26 EXTEND
27 INDEX ADRSDIF1
28 SU FIREDDB
29 EXTEND
30 BZMF SENSTEST # IF NOT: ARE UNBALANCED JETS PREFERRED?
31 MAXJETS CAF TWO # IF YES: INCREMENT ADDRESS LOCATOR AND
32 ADS ADRSDIF2 # SET SWITCH FOR JET SELECT LOGIC TO 4.
33 CAF FOUR # (ALWAYS DO THIS FOR P-AXIS)
34
35 SENSTEST CCS TJCALC # DOES TRANSLATION PREFER MIN JETS.
36 TCF SENSETYP # YES. USE MIN-JET PARAMETERS
37 TCF TJCALC
38
39 TJCALC TCF MAXJETS # NO. GET THE MAX-JET PARAMETERS.
40 TS NUMBERT # SET TO +0,1,4 FOR (U,V-AXES) JET SELECT.
41
42 # BEGINNING OF TJET CALCULATIONS:
43
44 CS EDOTSQ # SCALED AT PI(2)/2(8).
45 EXTEND
46 INDEX ADRSDIF2
47 MP 1/ANET1 # .5/ACC SCALED AT 2(6)/PI SEC(2)/RADIAN.
48
49 INDEX ADRSDIF1
50 AD FIREDDB # DEADBAND SCALED AT PI/4 RADIAN.
51 EXTEND
52 SU E # ATTITUDE ERROR SCALED AT PI/4 RADIAN.
53 TS FIREFCT # -E-.5(EDOTSQ)/ACC-DB AT PI/4 RADIAN.
54 EXTEND
55 BZMF ZON1,2,3
56
57 ZONE4,5 INDEX ADRSDIF1
58 CAE 1/ACOAST # .5/ACC SCALED AT 2(6)/PI WHERE
```

```
1
2 EXTEND # ACC = MAX(AMIN, AOS-).
3 MP # SCALED AT PI/2(8).
4 EDOTSQ # SCALED AT PI/4
5 AD # SCALED AT PI/4 POS. FOR NEG. INTERCEPT.
6 INDEX # TEST E+.5(EDOTSQ)/ACC+DB AT PI/4 RADIAN.
7 ADRSDIF1 # IF FUNCTION NEGATIVE, FIND TJET.
8 AD # IF FUNCTION POSITIVE, IN ZONE 4.
9 COASTDB
10 EXTEND
11 BZMF ZONE5
12
13 # ZONE 4 IS THE COAST REGION. HOWEVER, IF THE JETS ARE ON AND DRIVING TOWARD
14 # A. THE AXIS WITHIN + OR - (DB + FLAT) FOR DRIFTING FLIGHT, OR
15 # B. THE USUAL TARGET PARABOLA FOR POWERED FLIGHT
16 # THEN THE THRUSTERS ARE KEPT ON.
17
18 ZONE4 INDEX AXISCTR # IS THE CURRENT VALUE IN TJET NON-ZERO
19 CS TJETU # WITH SENSE OPPOSITE TO EDOT,
20 EXTEND # (I.E., ARE JETS ON AND FIRING TOWARD
21 MP ROTSENSE # THE DESIRABLE STATE).
22 EXTEND
23 BZMF COASTTJ # NO. COAST.
24
25 JETSON CCS FLAT # YES. IS THIS DRIFTING OR POWERED FLIGHT?
26 TCF DRIFT/ON # DRIFTING. GO MAKE FURTHER TEST.
27
28 CS FIREFCT # POWERED (OR ULLAGE). CAN TARGET PARABOLA
29 INDEX ADRSDIF1 # BE REACHED FROM THIS POINT IN THE
30 AD AXISDIST # PHASE PLANE?
31 EXTEND
32 BZMF COASTTJ # NO. SET TJET = 0.
33 TC Z123COMP # YES. CALCULATE TJET AS THOUGH IN ZONE 1
34 CAE FIREFCT # AFTER COMPUTING THE REQUIRED
35 TCF ZONE1 # PARAMETERS.
36
37 DRIFT/ON INDEX ADRSDIF1 # CAN TARGET STRIP OF AXIS BE REACHED FROM
38 CS FIREDDB # THIS POINT IN THE PHASE PLANE?
39 DOUBLE
40 AD FIREFCT
41 EXTEND
42 BZMF +3
43 CAF ZERO # NO. SET TJET = 0.
44 TCF RETURNTJ
45
46 TC Z123COMP # YES. CALCULATE TJET AS THOUGH IN ZONE 2
47 TCF ZONE2,3 # OR 3 AFTER COMPUTING REQUIRED VALUES.
48
49 ZONE5 TS L # TEMPORARILY STORE FUNCTION IN L.
50 CCS ROTSENSE # MODIFY ADRSDIF2 FOR ACCESSING 1/ANET2
51 TCF +4 # AND ACCFCTZ5, WHICH MUST BE PICKED UP
52 TC CCSHOLE # FROM THE NEXT LOWER REGISTER IF THE
53 CS TWO # (ACTUAL) ERROR RATE IS NEGATIVE.
```



```
1
2 ADS ADRSDIF2
3
4 +4 CAE L
5 EXTEND
6 INDEX ADRSDIF2 # TTOAXIS AND HH ARE THE PARAMETERS UPON
7 MP ACCFCTZ5 # WHICH THE APPROXIMATIONS TO TJET ARE
8 DDOUBL # ABASED.
9 DDOUBL
10 DXCH HH # DOUBLE PRECISION H SCALED AT 8 SEC(2).
11 INDEX ADRSDIF2
12 CAE 1/ANET2 # SCALED AT 2(7)/PI SEC(2)/RAD.
13 EXTEND
14 MP EDOT # SCALED AT PI/2(5)
15 TS TTOAXIS # SCALED AT 4 SEC.
16
17 # TEST WHETHER TJET GREATER THAN 50 MSEC.
18
19 EXTEND
20 MP -.05AT2 # H - .05 TTOAXIS - .00125 G.T. ZERO
21 AD HH # (SCALED AT 8 SEC(2)).
22 AD NEG2
23 EXTEND
24 BZMF FORMULA1
25
26 # TEST WHETHER TJET GREATER THAN 150 MSEC.
27
28 CAE TTOAXIS
29 EXTEND
30 MP -.15AT2 # H - .15 TTOAXIS - .01125 G.T. ZERO
31 AD HH # (SCALED AT 8 SEC(2))
32 AD -.0112A8
33 EXTEND
34 BZMF FORMULA2
35
36 # IF TJET GREATER THAN 150 MSEC, ASSIGN IT VALUE OF 250 MSEC, SINCE THIS
37 # IS ENOUGH TO ASSURE NO SKIP NEXT CSP (100 MSEC).
38
39 FULLTIME CAF BIT11 # 250 MSEC SCALED AT 4 SEC.
40
41 # RETURN TO CALLING PROGRAM WITH JET TIME SCALED AS TIME6 AND SIGNED.
42
43 RETURNTJ EXTEND # ALL BRANCHES TERMINATE HERE WITH TJET
44 MP ROTSENSE # (SCALED AT 4 SEC) IN THE ACCUMULATOR.
45 INDEX AXISCTR # ROTSENSE APPLIES SIGN AND CHANGES SCALE.
46 TS TJETU
47 EXTEND
48 INDEX AXISCTR
49 MP ACCSWU # SET SWITCH FOR JET SELECT IF ROTATION IS
50 CAE L
51 EXTEND # IN A SENSE FOR WHICH 1/ACCS HAS FORCED
52 BZMF +3 # A MAX-JET CALCULATION.
53 CAF FOUR
54
55
56
57
58
59
60
```

```
1
2 TS NUMBERT
3 TC HOLDQ # RETURN VIA SAVED Q.
4
5 # TJET = H/(.025 + TTOAXIS) FOR TJET LESS THAN 50 MSEC.
6
7 FORMULA1 CS -.025AT4 # .025 SEC SCALED AT 4.
8 AD TTOAXIS # SCALED AT 4 SECONDS.
9 DXCH HH # STORE DENOMINATOR IN FIRST WORD OF H,
10 EXTEND # WHICH NEED NOT BE PRESERVED. PICK UP
11 DV HH # DP H AND DIVIDE BY DENOMINATOR.
12 EXTEND
13 MP BIT14 # RESCALE TJET FROM 2 TO USUAL 4 SEC.
14 TCF CHKMINTJ # CHECK THAT TJET IS NOT LESS THAN MINIMUM
15
16 # TJET = (H + .00375)/(0.1 + TTOAXIS) FOR TJET GREATER THAN 50 MSEC.
17
18 FORMULA2 EXTEND
19 DCA .00375A8 # .00375 SEC(2) SCALED AT 8.
20 DAS HH # STORE NUMERATOR IN DP H, WHICH NEED NOT
21 # BE PRESERVED.
22 CAE TTOAXIS # SCALED AT 4 SEC.
23 AD .1AT4 # 0.1 SEC SCALED AT 4.
24 DXCH HH # STORE DENOMINATOR IN FIRST WORD OF H,
25 EXTEND # WHICH NEED NOT BE PRESERVED. PICK UP
26 DV HH # DP NUMERATOR AND DIVIDE BY DENOMINATOR
27 EXTEND
28 MP BIT14 # RESCALE TJET FROM 2 TO USUAL 4 SEC.
29 TCF RETURNTJ # END SUBROUTINE.
30
31 # SUBROUTINIZED COMPUTATIONS REQUIRED FOR ALL ENTRIES INTO CODING FOR ZONES 1, 2, AND 3.
32 # REACHED BY TC FROM 3 POINTS IN TJETLAW.
33
34 Z123COMP CS ROTSENSE # USED IN RETURNTJ SECTION TO RESCALE TJET
35 TS ROTSENSE # AS TIME6 AND GIVE IT PROPER SIGN.
36 CAE EDOT # SCALED AT PI/2(5) RAD/SEC.
37 EXTEND
38 INDEX ADRSDIF2
39 MP 1/ANET1 # SCALED AT 2(7)/PI SEC(2)/RAD.
40 TS TTOAXIS # STORE TIME-TO-AXIS SCALED AT 4 SECONDS.
41 AD -TJMAX
42 EXTEND # IS TIME TO AXIS LESS THAN 150 MSEC.
43 BZMF +2
44 TCF FULLTIME # NO. FIRE JETS, DO NOT CALCULATE TJET.
45 RETURN # YES. GO ON TO FIND TJET
46
47 ZON1,2,3 TC Z123COMP # SUBROUTINIZED PREPARATION FOR ZONE1,2,3.
48
49 # IF THE (NEG) DISTANCE BEYOND PARABOLA IS LESS THAN FLAT, USE SPECIAL
50 # LOGIC TO ACQUIRE MINIMUM IMPULSE LIMIT CYCLE. DURING POWERED FLIGHT
```

# OR ULLAGE, FLAT = 0

|        |         |                       |
|--------|---------|-----------------------|
| CAE    | FIREFCT | # SCALED AT PI/4 RAD. |
| AD     | FLAT    |                       |
| EXTEND |         |                       |

|      |       |                         |
|------|-------|-------------------------|
| BZMF | ZONE1 | # NOT IN SPECIAL ZONES. |
|------|-------|-------------------------|

# FIRE FOR AXIS OR, IF CLOSE, FIRE MINIMUM IMPULSE. IF ON AXIS, COAST.

|         |        |          |                                            |
|---------|--------|----------|--------------------------------------------|
| ZONE2,3 | CS     | ZONE3LIM | # HEIGHT OF MIN-IMPULSE ZONE SET BY 1/ACCS |
|         | AD     | TTOAXIS  | # 35 MSEC IN DRIFTING FLIGHT               |
|         | EXTEND |          | # ZERO WHEN TRYING TO ENTER GTS CONTROL.   |

|       |      |         |                 |
|-------|------|---------|-----------------|
| ZONE2 | BZMF | ZONE3   |                 |
|       | CAE  | TTOAXIS | # FIRE TO AXIS. |

|       |     |          |                                            |
|-------|-----|----------|--------------------------------------------|
| ZONE3 | TCF | RETURNTJ |                                            |
|       | CCS | EDOT     | # CHECK IF EDOT IS ZERO.                   |
|       | CAF | BIT6     | # FIRE A ONE-JET MINIMUM IMPULSE.          |
|       | TCF | RETURNTJ | # TJET = +0.                               |
|       | TC  | CCSHOLE  | # CANNOT BE BECAUSE NEG EDOT COMPLEMENTED. |
|       | TCF | RETURNTJ | # TJET = +0.                               |

|       |        |          |                                 |
|-------|--------|----------|---------------------------------|
| ZONE1 | EXTEND |          |                                 |
|       | INDEX  | ADRSDIF1 |                                 |
|       | SU     | AXISDIST | # SCALED AT PI/4 RAD.           |
|       | EXTEND |          |                                 |
|       | INDEX  | ADRSDIF2 |                                 |
|       | MP     | ACCFCTZ1 | # SCALED AT 2(7)/PI SEC(2)/RAD. |

|        |    |                                          |  |
|--------|----|------------------------------------------|--|
| DDOUBL |    |                                          |  |
| DDOUBL |    |                                          |  |
| DXCH   | HH | # DOUBLE PRECISION H SCALED AT 8 SEC(2). |  |

# TEST WHETHER TOTAL TIME REQUIRED GREATER THAN 150 MSEC:

|   |                       |     |     |                                 |
|---|-----------------------|-----|-----|---------------------------------|
| # |                       | $2$ |     | $2$                             |
| # | IS .5(.150 - TTOAXIS) |     | - H | NEGATIVE (SCALED AT 8 SECONDS ) |

|     |         |                                |
|-----|---------|--------------------------------|
| CAE | TTOAXIS | # TTOAXIS SCALED AT 4 SECONDS. |
| AD  | -TJMAX  | # -.150 SECOND SCALED AT 4.    |

|        |    |                                      |  |
|--------|----|--------------------------------------|--|
| EXTEND |    |                                      |  |
| SQUARE |    |                                      |  |
| EXTEND |    |                                      |  |
| SU     | HH | # HIGH WORD OF H SCALED AT 8 SEC(2). |  |

|        |          |                                 |  |
|--------|----------|---------------------------------|--|
| EXTEND |          |                                 |  |
| BZMF   | FULLTIME | # YES. NEED NOT CALCULATE TJET. |  |

# TEST WHETHER TIME BEYOND AXIS GREATER THAN 50 MSEC TO DETERMINE WHICH APPROXIMATION TO USE.

|        |          |  |
|--------|----------|--|
| CAE    | HH       |  |
| AD     | NEG2     |  |
| EXTEND |          |  |
| BZMF   | FORMULA3 |  |

# TJET = H/0.1 + TTOAXIS + .0375 FOR APPROXIMATION OVER MORE THAN 50 MSEC.

|        |          |                                          |
|--------|----------|------------------------------------------|
| CAF    | .1AT2    | # STORE .1 SEC SCALED AT 2 FOR DIVISION. |
| DXCH   | HH       | # DP H SCALED AT 8 SEC(2) NEED NOT BE    |
| EXTEND |          | # PRESERVED.                             |
| DV     | HH       | # QUOTIENT SCALED AT 4 SECONDS.          |
| AD     | TTOAXIS  | # SCALED AT 4 SEC.                       |
| AD     | .0375AT4 | # .0375 SEC SCALED AT 4.                 |
| TCF    | RETURNTJ | # END COMPUTATION.                       |

# TJET - H/.025 + TTOAXIS FOR APPROXIMATION OVER LESS THAN 50 MSEC.

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| FORMULA3 | CS     | -.025AT2 | # STORE +.25 SEC SCALED AT 2 FOR DIVISION |
|          | DXCH   | HH       | # PICK UP DP H AT 8, WHICH NEED NOT BE    |
|          | EXTEND |          | # PRESERVED.                              |
|          | DV     | HH       | # QUOTIENT SCALED AT 4 SECONDS.           |
|          | AD     | TTOAXIS  | # SCALED AT 4 SEC.                        |

# IF COMPUTED JET TIME IS LESS THAN TJMIN, TJET IS SET TO ZERO.  
# MINIMUM IMPULSES REQUIRED IN ZONE 3 ARE NOT SUBJECT TO THIS CONSTRAINT, NATURALLY.

|          |        |          |                                           |
|----------|--------|----------|-------------------------------------------|
| CHKMINTJ | AD     | -TJMIN   | # IS COMPUTED TIME LESS THAN THE MINIMUM. |
|          | EXTEND |          |                                           |
|          | BZMF   | COASTTJ  | # YES, SET TIME TO ZERO.                  |
|          | AD     | TJMIN    | # NO, RESTORE COMPUTED TIME.              |
|          | TCF    | RETURNTJ | # END COMPUTATION.                        |

# \*\*\* ROUGHLAW \*\*\*

#

# BEFORE ENTRY TO RUFLAW:

# 1. INDEXERS ADRSDIF1 AND ADRSDIF2 ARE SET ON BASIS OF AXIS, AND SIGN OF EDOT.

# 2. IF EDOT WAS NEGATIVE, E AND EDOT ARE ROTATED INTO UPPER HALF-PLANE AND ROTSENSE IS MADE NEGATIVE.

# 3. E IS SCALED AT PI RADIANS AND EDOT AT PI/4 RAD/SEC.

# (EXCEPT THE RUFLAW3 ENTRY WHEN E IS AT PI/4)

#

# RUFLAW1: ERROR MORE NEGATIVE THAN PI/16 RAD. FIRE TO A RATE OF 6.5 DEG/SEC (IF JET TIME EXCEEDS 20 MSEC.).

# RUFLAW2: ERROR MORE POSITIVE THAN PI/16 RAD. FIRE TO AN OPPOSING RATE OF 6.5 DEG/SEC.

# RUFLAW3: ERROR RATE GREATER THAN PI/32 RAD/SEC AND ERROR WITHIN BOUNDS. COAST IF BELOW FIREFCT, FIRE IF ABOVE

RUFLAW1

CS

RUF RATE

# DECREMENT EDOT BY .1444 RAD/SEC AT PI/4

ADS

EDOT

# WHICH IS THE TARGET RATE

EXTEND

BZMF

SMAL RATE

# BRANCH IF RATE LESS THAN TARGET.

TC

RUF SETUP

# REVERSE ROTSENSE AND INDICATE MAX JETS.

CAE

EDOT

# PICK UP DESIRED RATE CHANGE.

RUFLAW12

EXTEND

# COMPUTE TJET

INDEX

ADRSDIF2

# = (DESIRED RATE CHANGE)/(2-JET ACCEL.)

MP

1/ANET1 +2

AD

-1/8

# IF TJET, SCALED AT 32 SEC, EXCEEDS

EXTEND

# 4 SECONDS, SET TJET TO TJMAX.

BZMF

+2

TCF

FULLTIME

EXTEND

BZF

FULLTIME

AD

BIT12

# RESTORE COMPUTED TJET TO ACCUMULATOR

DAS

A

DAS

A

DAS

A

# RESCALED TJET AT 4 SECONDS.

TCF

CHKMINTJ

# RETURN AS FROM FINELAW.

SMAL RATE

TC

RUF SETUP +2

# SET NUMBERT AND FIREFCT FOR MAXIMUM JETS

CCS

ROTSSENSE

CAF

ONE

# MODIFY INDEXER TO POINT TO 1/ANET

TCF

+2

# CORRESPONDING TO THE PROPER SENSE.

CAF

NEGONE

ADS

ADRSDIF2

CS

EDOT

# (.144 AT PI/4 - EDOT) = DESIRED RATE CHNG.

TCF

RUFLAW12

RUFLAW2

TC

RUF SETUP

# REVERSE ROTSENSE AND INDICATE MAX JETS.

CAF

RUF RATE

AD

EDOT

# (.144 AT PI/4 + EDOT) = DESIRED RATE CHNG.

TS

A

# IF OVERFLOW SKIP, FIRE FOR FULL TIME.

TCF

RUFLAW12

# OTHERWISE, COMPUTE JET TIME.

TCF

FULLTIME

```
1 RUFLAW3 TC RUFSETUP # EXECUTE COMMON RUFLAW SUBROUTINE.
2 INDEX ADRSDIF1
3
4 CS FIREDB # CALCULATE DISTANCE FROM SWITCH CURVE
5 AD E # 1/ANET1*EDOT*EDOT +E - FIREDB = 0
6 EXTEND # # SCALED AT 4 PI RADIANS
7
8 MP BIT11
9 XCH EDOT
10 EXTEND
11 SQUARE
12 EXTEND
13 INDEX ADRSDIF1
14 MP 1/ANET1 +2
15 AD EDOT
16 EXTEND
17 BZMF COASTTJ # COAST IF BELOW IT.
18 TCF FULLTIME # FIRE FOR FULL PERIOD IF ABOVE IT.
19
20 # SUBROUTINE USED IN ALL ENTRIES TO ROUGHLAW.
21
22 RUFSETUP CS ROTSENSE # REVERSE ROTSENSE WHEN ENTER HERE.
23 +2 TS ROTSENSE
24 CAF FOUR # REQUIRE MAXIMUM (2) JETS IN U,V-AXES.
25 TS NUMBERT
26 CAF NEGMAX # SUGGEST MAXIMUM (4) JETS IN P-AXIS.
27 TS FIREFCT
28 TC Q
29
30 # CONSTANTS FOR TJETLAW
31
32 AXISDIFF DEC -16 # AXISDIFF(INDEX) = NUMBER OF REGISTERS
33 DEC +0 # BETWEEN STORED 1/ACCS PARAMETERS FOR
34 DEC 16 # THE INDEXED AXIS AND THE U-AXIS.
35
36 SENSOR OCT 14400 # RATIO OF TJET SCALING WITHIN TJETLAW
37 # # (4 SEC) TO SCALING FOR T6 (10.24 SEC).
38
39 -3DEG DEC -.06667 # -3.0 DEGREES SCALED AT 45.
40 -.0112A8 DEC -.00141 # -.01125 SEC(2) SCALED AT 8.
41 .1AT4 DEC .025 # 0.1 SECOND SCALED AT 4.
42 .1AT2 DEC .05 # .1 SEC SCALED AT 2.
43 .0375AT4 DEC .00938 # .0375 SEC SCALED AT 4.
44 -.025AT2 DEC -.0125 # -.025 SEC SCALED AT 2.
45 -.025AT4 DEC -.00625
46 -.05AT2 DEC -.025
47 -.15AT2 DEC -.075
48 .00375A8 2DEC .00375 B-3
49
50 -TJMAX DEC -.0375 # LARGEST CALCULATED TIME. .150 SEC AT 4.
51 TJMIN DEC .005 # SMALLEST ALLOWABLE TIME. .020 SEC AT 4.
52 -TJMIN DEC -.005
53 RUFRATE DEC .1444 # CORRESPONDS TO TARGET RATE OF 6.5 DEG/S.
```

```
1
2 EBANK= NO.UJETS
3 BANK 16
4 SETLOC DAPS1
5 BANK
6
7 COUNT* $$/DAP
8
9 RATELOOP CA TWO
10 TS DAPTEMP6
11 DOUBLE
12 TS Q
13 INDEX DAPTEMP6
14 CCS TJP
15 TCF +2
16 TCF LOOPRATE
17 AD -100MST6
18 EXTEND
19 BZMF SMALLTJU
20 INDEX DAPTEMP6
21 CCS TJP
22 CA -100MST6
23 TCF +2
24 CS -100MST6
25 INDEX DAPTEMP6
26 ADS TJP
27 INDEX DAPTEMP6
28 CCS TJP
29 CS -100MS # 0.1 AT 1
30 TCF +2
31 CA -100MS
32 LOOPRATE EXTEND
33 INDEX DAPTEMP6
34 MP NO.PJETS
35 CA L
36 INDEX DAPTEMP6
37 TS DAPTEMP1 # SIGNED TORQUE AT 1 JET-SEC FOR FILTER
38 EXTEND
39 MP BIT10 # RESCALE TO 32; ONE BIT ABOUT 2 JET-MSEC
40 EXTEND
41 BZMF NEGTOCK
42 STORTORK INDEX Q # INCREMENT DOWNLIST REGISTER.
43 ADS DOWNTORK # NOTE: NOT INITIALIZED; OVERFLOWS.
44
45 CCS DAPTEMP6
46 TCF RATELOOP +1
47 TCF ROTORQUE
48 SMALLTJU CA ZERO
49 INDEX DAPTEMP6
50 XCH TJP
51 EXTEND
52
53
54
55
56
57
58
59
60
```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65
- 66
- 67
- 68
- 69
- 70
- 71
- 72
- 73
- 74
- 75
- 76
- 77
- 78
- 79
- 80
- 81
- 82
- 83
- 84
- 85
- 86
- 87
- 88
- 89
- 90
- 91
- 92
- 93
- 94
- 95
- 96
- 97
- 98
- 99
- 100

14121HE

1



```
1 BANK 21
2 EBANK= QDIFF
3
4 SETLOC DAPS4
5 BANK
6
7 COUNT* $$/DAPGT
8
9 # CONTROL REACHES THIS POINT UNDER EITHER OF THE FOLLOWING TWO CONDITIONS ONCE THE DESCENT ENGINE AND THE DIGITAL
10 # AUTOPILOT ARE BOTH ON:
11 # A) THE TRIM GIMBAL CONTROL LAW WAS ON DURING THE PREVIOUS Q,R-AXIS TIME5 INTERRUPT (OR THE DAPIDLER
12 # INITIALIZATION WAS SET FOR TRIM GIMBAL CONTROL AND THIS IS THE FIRST PASS), OR
13 # B) THE Q,R-AXES RCS AUTOPILOT DETERMINED THAT THE VEHICLE WAS ENTERING (OR HAD JUST ENTERED) A COAST
14 # ZONE WITH A SMALL OFFSET ANGULAR ACCELERATION.
15 # GTS IS THE ENTRY TO THE GIMBAL TRIM SYSTEM FOR CONTROLLING ATTITUDE ERRORS AND RATES AS WELL AS ACCELERATIONS.
16
17 GTS CAF NEGONE # MAKE THE NEXT PASS THROUGH THE DAP BE
18 TS COTROLER # THROUGH RCS CONTROL,
19 CAF FOUR # AND ENSURE THAT IT IS NOT A SKIP.
20 TS SKIPU
21 TS SKIPV
22
23 CAF TWO
24 TS INGTS # SET INDICATOR OF GTS CONTROL POSITIVE.
25 TS QGIMTIMR # SET TIMERS TO 200 MSEC TO AVOID BOTH
26 TS RGIMTIMR # RUNAWAY AND INTERFERENCE BY NULLING.
27
28 # THE DRIVE SETTING ALGORITHM
29 #
30 # DEL = SGN(OMEGA + ALPHA*ABS(ALPHA)/(2*K))
31 #
32 # NEGUSUM = ERROR*K + ALPHA*(DEL*OMEGA + ALPHA / (3*K)) + DEL*K2 (DEL*OMEGA + ALPHA / (2*K))1/2
33 #
34 # DRIVE = -SGN(NEGUSUM)
35
36 CA SR # SAVE THE SR. SHIFT IT LEFT TO CORRECT
37 AD A # FOR THE RIGHT SHIFT DUE TO EDITING.
38 TS SAVESR
39
40 GTSGO+ON CAF TWO # SET INDEXER FOR R-AXIS CALCULATIONS.
41 TCF GOQTRIMG +1
42
43 GOQTRIMG CAF ZERO # SET INDEXER FOR Q-AXIS CALCULATIONS
44 TS QRCNTR
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

# RSB 2009 -----  
# EVERYTHING BETWEEN THIS LINE AND THE SIMILAR LINE BELOW WAS SIMPLY FILLED-IN  
# AS-IS FROM LUMINARY 131, AND THEN VERIFIED TO ASSEMBLE TO THE PROPER BINARY  
# VALUES. THIS AREA IS BLANK ON THE LUMINARY 099 PRINT-OUT, AS IF THE PRINTER  
# RIBBON HAD RUN OUT.

INDEX QRCNTR # AOS SCALED AT PI/2  
CA AOSQ  
EXTEND

MP BIT2 # RESCALE AOS TO PI/4  
EXTEND

BZF GTSQAXIS -3 # USE FULL SCALE FOR LARGER AOS ESTIMATES.

INDEX A  
CS LIMITS # LIMITS +1 CONTAINS NEGMAX.  
XCH L # LIMITS -1 CONTAINS POSMAX.

CCS QRCNTR # PICK UP RATE FOR THIS AXIS. RATE CELLS  
INDEX A # USE ADJACENT, NOT SEPARATED. AT PI/4

GTSQAXIS

CA EDOTQ  
DXCH WCENTRAL

INDEX QRCNTR # COLLECT K FOR THIS AXIS  
CA KQ

TS KCENTRAL

EXTEND # CONTROL AUTHORITY ZERO. AVOID DRIVING  
BZF POSDRIVE +1 # ENGINE BELL TO THE STOPS.

INDEX QRCNTR # QDIFF, RDIFF ARE STORED IN D.P.  
CAE QDIFF

ALGORITHM

EXTEND # Q(R)DIFF IS THETA (ERROR) SCALED AT PI.  
MP KCENTRAL # FORM K\*ERROR AT PI(2)/2(8), IN D.P.

LXCH K2THETA  
EXTEND

MP BIT5 # RESCALE TO 4\*PI(2)  
DXCH K2THETA  
EXTEND

MP BIT5 # FIRST TERM OF NEGUSUM IN K2THETA.  
ADS K2THETA +1 # NO CARRY NEEDED D.P. AT 4\*PI(2)

CS ACENTRAL # FORM ALPHA(2)/(2\*K) AT 16\*PI, IN D.P.,  
EXTEND # LIMITING QUOTIENT TO AVOID OVERFLOW.  
MP BIT14 # -ALPHA/2 IN A, SCALED AT PI/4

EXTEND  
MP ACENTRAL # -ALPHA(2)/2 IN A,L, SCALED AT PI(2)/16  
AD KCENTRAL

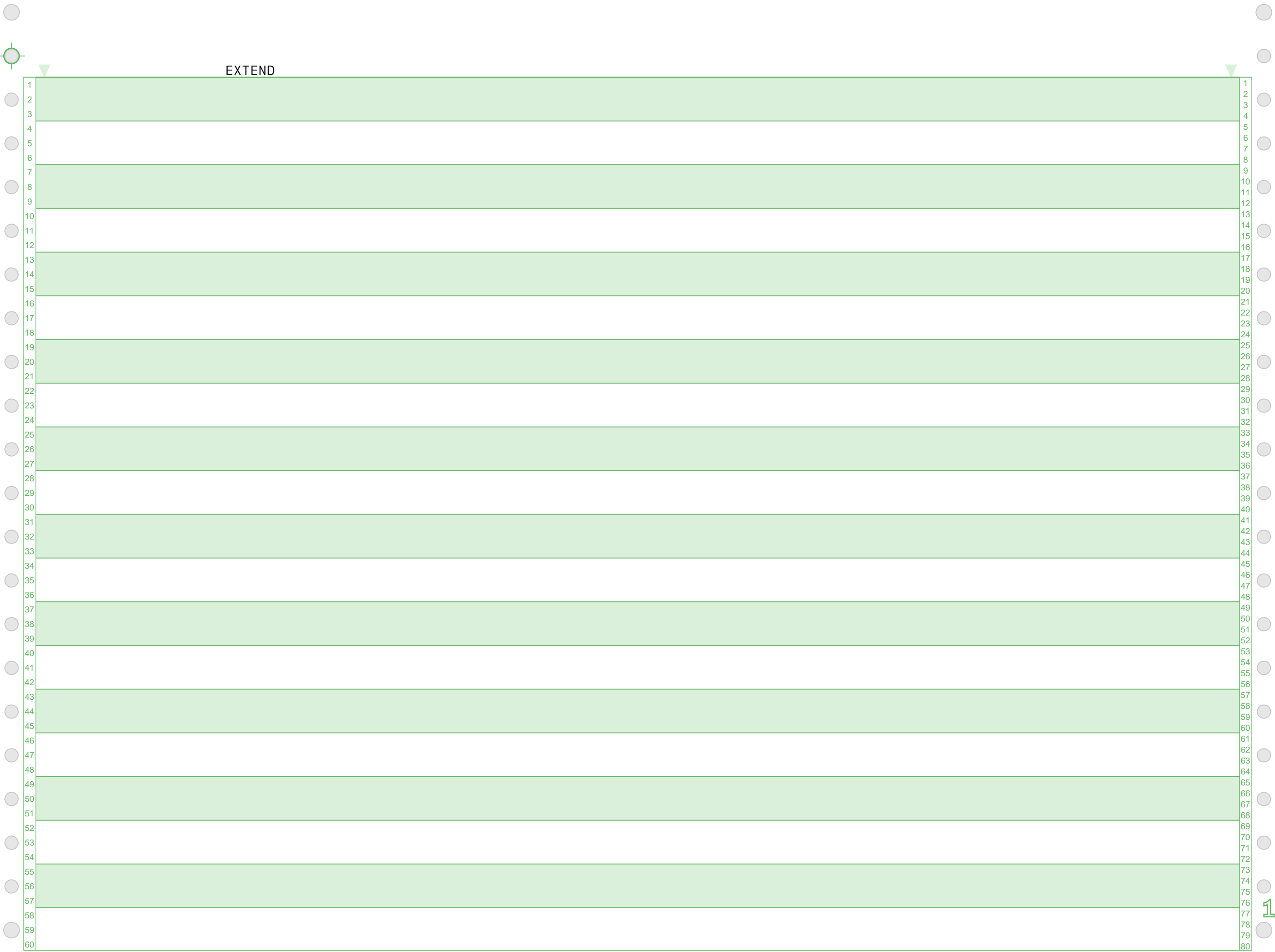
EXTEND  
BZMF HUGEQUOT # K-ALPHA(2)/2 SHOULD BE PNZ FO DIVISION

EXTEND  
DCS A # ALPHA(2)/2 - K  
AD KCENTRAL

# RSB 2009 -----

EXTEND  
DV KCENTRAL # HIGH ORDER OF QUOTIENT.

XCH A2CNTRAL  
CA L # SHIFT UP THE REMAINDER.  
LXCH 7 # ZERO LOW-ORDER DIVIDEND.



|    |           |        |             |                                           |
|----|-----------|--------|-------------|-------------------------------------------|
| 1  |           |        |             |                                           |
| 2  |           | DV     | KCENTRAL    |                                           |
| 3  |           | XCH    | A2CNTRAL +1 | # QUOTIENT STORED AT 16*PI , D.P.         |
| 4  |           | TCF    | HAVEQUOT    |                                           |
| 5  |           |        |             |                                           |
| 6  | HUGEQUOT  | CA     | POSMAX      |                                           |
| 7  |           | TS     | L           |                                           |
| 8  |           | DXCH   | A2CNTRAL    | # LIMITED QUOTIENT STORED AT 16*PI, D.P.  |
| 9  |           |        |             |                                           |
| 10 | HAVEQUOT  | CA     | WCENTRAL    |                                           |
| 11 |           | EXTEND |             |                                           |
| 12 |           | MP     | BIT9        | # RESCALE OMEGA AT 16*PI IN D.P.          |
| 13 |           | DXCH   | K2CNTRAL    | # LOWER WORD OVERLAYS OMEGA IN WCENTRAL   |
| 14 |           |        |             |                                           |
| 15 |           | EXTEND |             |                                           |
| 16 |           | DCA    | K2CNTRAL    |                                           |
| 17 |           | DXCH   | FUNCTION    |                                           |
| 18 |           |        |             |                                           |
| 19 |           | CA     | ACENTRAL    | # GET ALPHA*ABS(ALPHA)/(2*K)              |
| 20 |           | EXTEND |             |                                           |
| 21 |           | BZMF   | +4          |                                           |
| 22 |           |        |             |                                           |
| 23 |           | EXTEND |             |                                           |
| 24 |           | DCA    | A2CNTRAL    |                                           |
| 25 |           | TCF    | +3          |                                           |
| 26 |           |        |             |                                           |
| 27 |           | EXTEND |             |                                           |
| 28 |           | DCS    | A2CNTRAL    |                                           |
| 29 |           |        |             |                                           |
| 30 |           | DAS    | FUNCTION    | # OMEGA + ALPHA*ABS(ALPHA)/(2*K) AT 16*PI |
| 31 |           |        |             |                                           |
| 32 |           | CCS    | FUNCTION    | # DEL = +1 FOR FUNCT1 GREATER THAN ZERO.  |
| 33 |           | TCF    | POSFUNCT1   | # OTHERWISE DEL = -1                      |
| 34 |           | TCF    | +2          |                                           |
| 35 |           | TCF    | NEGFUNCT1   |                                           |
| 36 |           |        |             |                                           |
| 37 |           | CCS    | FUNCTION +1 | # USE LOW ORDER WORD SINCE HIGH IS ZERO   |
| 38 | POSFUNCT1 | CAF    | BIT1        |                                           |
| 39 |           | TCF    | +2          |                                           |
| 40 | NEGFUNCT1 | CS     | BIT1        |                                           |
| 41 |           | TS     | DEL         |                                           |
| 42 |           |        |             |                                           |
| 43 |           | CCS    | DEL         | # REPLACE OMEGA BY DEL*OMEGA              |
| 44 |           | TCF    | FUNCT2      | # POSITIVE DEL VALUE. PROCEED.            |
| 45 |           | TCF    | DEFUNCT     |                                           |
| 46 |           | TCF    | NEGFUNCT2   |                                           |
| 47 |           |        |             |                                           |
| 48 | DEFUNCT   | TS     | K2CNTRAL    |                                           |
| 49 |           | TS     | K2CNTRAL +1 |                                           |
| 50 |           | TCF    | FUNCT2      |                                           |
| 51 |           |        |             |                                           |
| 52 |           |        |             |                                           |
| 53 |           |        |             |                                           |
| 54 |           |        |             |                                           |
| 55 |           |        |             |                                           |
| 56 |           |        |             |                                           |
| 57 |           |        |             |                                           |
| 58 |           |        |             |                                           |
| 59 |           |        |             |                                           |
| 60 |           |        |             |                                           |

```
1 NEG1/3 DEC -.33333
2
3
4 NEGFNCT2 EXTEND
5 DCS K2CNTRAL
6 DXCH K2CNTRAL
7
8 FUNCT2 EXTEND
9 DCA A2CNTRAL
10 DAS K2CNTRAL # DEL*OMEGA + ALPHA(2)/(2*K) AT 16*PI,D.P.
11
12 FUNCT3 CA A2CNTRAL
13 EXTEND
14 MP NEG1/3
15 DXCH A2CNTRAL
16 CA L
17 EXTEND
18 MP NEG1/3
19 ADS A2CNTRAL +1
20 TS L
21 TCF +2 # A2CNTRAL NOW CONTAINS -ALPHA(2)/(6*K),
22 ADS A2CNTRAL # SCALED AT 16*PI, IN D.P.
23
24 EXTEND
25 DCA K2CNTRAL # DEL*OMEGA + ALPHA(2)/(3*K) IN A2CNTRAL,
26 DAS A2CNTRAL # SCALED AT 16*PI, D.P.
27
28 CA A2CNTRAL
29 EXTEND
30 MP ACENTRAL
31 DAS K2THETA
32 CA A2CNTRAL +1
33 EXTEND
34 MP ACENTRAL # ACENTRAL MAY NOW BE OVERLAID.
35 ADS K2THETA +1
36 TS L
37 TCF +2 # TWO TERMS OF NEGUSUM ACCUMULATED, SO FAR
38 ADS K2THETA # SCALED AT 4*PI(2), IN D.P.
39
40 GETROOT CA K2CNTRAL # K*(DEL*OMEGA + ALPHA(2)/(2*K)) IS THE
41 EXTEND # TERM FOR WHICH A SQUARE ROOT IS NEEDED.
42 MP KCENTRAL # K AT PI/2(8)
43 DXCH FUNCTION
44 CA K2CNTRAL +1
45 EXTEND
46 MP KCENTRAL
47 ADS FUNCTION +1
48 TS L
49 TCF +2
50 ADS FUNCTION # DESIRED TERM IN FUNCTION, AT PI(2)/16
51
52
53
54
55
56
57
58
59
60
```

```
 CCS DEL
 TCF RSTOFGTS
 TCF NEGUSUM
 TCF NEGATE
 TCF NEGUSUM
```

```
NEGATE EXTEND
 DCS K2CNTRAL
 DXCH K2CNTRAL
 TCF RSTOFGTS
```

```
 BANK 16
 EBANK= NEGUQ
 SETLOC DAPS1
 BANK
```

```
THE WRCHN12 SUBROUTINE SETS BITS 9,10,11,12 OF CHANNEL 12 ON THE BASIS OF THE CONTENTS OF NEGUQ,NEGUR WHICH ARE
THE NEGATIVES OF THE DESIRED ACCELERATION CHANGES. ACDT+C12 SETS Q(R)ACCDOT TO REFLECT THE NEW DRIVES.
#
WARNING: ACDT+C12 AND WRCHN12 MUST BE CALLED WITH INTERRUPT INHIBITED.
```

```
BGIM OCTAL 07400
CHNL12 EQUALS ITEMP6
ACDT+C12 CS NEGUQ
 EXTEND
 MP ACCDOTQ # GIMBAL DRIVE REQUESTS.
 LXCH QACCDOT
 CS NEGUR
 EXTEND
 MP ACCDOTR
 LXCH RACCDOT
```

```
 CCS NEGUQ
 CAF BIT10
 TCF +2
 CAF BIT9
 TS CHNL12
```

```
 CCS NEGUR
 CAF BIT12
 TCF +2
 CAF BIT11
 ADS CHNL12 # (STORED RESULT NOT USED AT PRESENT)
```

```
 CS BGIM
 EXTEND
 RAND CHAN12
 AD CHNL12
 EXTEND
 WRITE CHAN12
```



|    |  |        |                     |    |
|----|--|--------|---------------------|----|
| 1  |  |        |                     | 1  |
| 2  |  | CS     | CALLGMBL            | 2  |
| 3  |  | MASK   | RCSFLAGS            | 3  |
| 4  |  | TS     | RCSFLAGS            | 4  |
| 5  |  |        |                     | 5  |
| 6  |  | TC     | Q                   | 6  |
| 7  |  |        | # RETURN TO CALLER. | 7  |
| 8  |  | BANK   | 21                  | 8  |
| 9  |  | EBANK= | QDIFF               | 9  |
| 10 |  | SETLOC | DAPS4               | 10 |
| 11 |  | BANK   |                     | 11 |
| 12 |  |        |                     | 12 |
| 13 |  |        |                     | 13 |
| 14 |  |        |                     | 14 |
| 15 |  |        |                     | 15 |
| 16 |  |        |                     | 16 |
| 17 |  |        |                     | 17 |
| 18 |  |        |                     | 18 |
| 19 |  |        |                     | 19 |
| 20 |  |        |                     | 20 |
| 21 |  |        |                     | 21 |
| 22 |  |        |                     | 22 |
| 23 |  |        |                     | 23 |
| 24 |  |        |                     | 24 |
| 25 |  |        |                     | 25 |
| 26 |  |        |                     | 26 |
| 27 |  |        |                     | 27 |
| 28 |  |        |                     | 28 |
| 29 |  |        |                     | 29 |
| 30 |  |        |                     | 30 |
| 31 |  |        |                     | 31 |
| 32 |  |        |                     | 32 |
| 33 |  |        |                     | 33 |
| 34 |  |        |                     | 34 |
| 35 |  |        |                     | 35 |
| 36 |  |        |                     | 36 |
| 37 |  |        |                     | 37 |
| 38 |  |        |                     | 38 |
| 39 |  |        |                     | 39 |
| 40 |  |        |                     | 40 |
| 41 |  |        |                     | 41 |
| 42 |  |        |                     | 42 |
| 43 |  |        |                     | 43 |
| 44 |  |        |                     | 44 |
| 45 |  |        |                     | 45 |
| 46 |  |        |                     | 46 |
| 47 |  |        |                     | 47 |
| 48 |  |        |                     | 48 |
| 49 |  |        |                     | 49 |
| 50 |  |        |                     | 50 |
| 51 |  |        |                     | 51 |
| 52 |  |        |                     | 52 |
| 53 |  |        |                     | 53 |
| 54 |  |        |                     | 54 |
| 55 |  |        |                     | 55 |
| 56 |  |        |                     | 56 |
| 57 |  |        |                     | 57 |
| 58 |  |        |                     | 58 |
| 59 |  |        |                     | 59 |
| 60 |  |        |                     | 60 |

```
1 # SUBROUTINE TIMEGMBL: MOD 0, OCTOBER 1967, CRAIG WORK
2 #
3 # TIMEGMBL COMPUTES THE DRIVE TIME NEEDED FOR THE TRIM GIMBAL TO POSITION THE DESCENT ENGINE NOZZLE SO AS TO NULL
4 # THE OFFSET ANGULAR ACCELERATION ABOUT THE Q (OR R) AXIS. INSTEAD OF USING AOSQ(R), TIMEGMBL USES .4*AOSQ(R),
5 # SCALED AT PI/8. FOR EACH AXIS, THE DRIVE TIME IS COMPUTED AS ABS(ALPHA/ACCDOT). A ZERO
6 # ALPHA OR ACCDOT OR A ZERO QUOTIENT TURNS OFF THE GIMBAL DRIVE IMMEDIATELY. OTHERWISE, THE GIMBAL IS TURNED ON
7 # DRIVING IN THE CORRECT DIRECTION. THE Q(R)GIMTIMR IS SET TO TERMINATE THE DRIVE AND Q(R)ACCDOT
8 # IS STORED TO REFLECT THE NEW ACCELERATION DERIVATIVE. NEGUQ(R) WILL CONTAIN +1,+0,-1 FOR A Q(R)ACCDOT VALUE
9 # WHICH IS NEGATIVE, ZERO, OR POSITIVE.
10 #
11 # INPUTS: AOSQ,AOSR, SCALED AT PI/2, AND ACCDOTQ, ACCDOTR AT PI/2(7). PI/2(7).
12 #
13 # OUTPUTS: NEW GIMBAL DRIVE BITS IN CHANNEL 12,NEGUQ,NEGUR,QACCDOT AND RACCDOT, THE LAST SCALED AT PI/2(7).
14 # Q(R)GIMTIMR WILL BE SET TO TIME AND TERMINATE GIMBAL DRIVE(S)
15 #
16 # DEBRIS: A,L,Q, ITEMPS 2,3,6, RUPTREG2 AND ACDT+C12 DEBRIS.
17 #
18 # EXITS: VIA TC Q.
19 #
20 # ALARMS, ABORTS, : NONE
21 #
22 # SUBROUTINES: ACDT+C12, IBNKCALL
23 #
24 # WARNING: THIS SUBROUTINE WRITES INTO CHANNEL 12 AND USES THE ITEMPS. THEREFORE IT MAY ONLY BE CALLED WITH
25 # INTERRUPT INHIBITED.
26 #
27 # ERASABLE STORAGE CONFIGURATION (NEEDED BY THE INDEXING METHODS):
28 #
29 # NEGUQ ERASE +2 # NEGATIVE OF Q-AXIS GIMBAL DRIVE
30 # (SPWORD) EQUALS NEGUQ +1 # ANY S.P. ERASABLE NUMBER, NOW THRSTCMD
31 # NEGUR ERASE +2 # NEGATIVE OF R-AXIS GIMBAL DRIVE
32 # ACCDOTQ ERASE +2 # Q-JERK TERM SCALED AT PI/2(7) RAD/SEC(3)
33 # (SPWORD) EQUALS ACCDOTQ +1 # ANY S.P. ERASABLE NUMBER NOW QACCDOT
34 # ACCDOTR ERASE +2 # R-JERK TERM SCALED AT PI/2(7) RAD/SEC(3)
35 # EQUALS ACCDOTQ +2 # ACCDOTQ,ACCDOTR ARE MAGNITUDES.
36 # AOSQ ERASE +4 # Q-AXIS ACC., D.P. AT PI/2 R/SEC(2)
37 # AOSR EQUALS AOSQ +2 # R-AXIS ACCELERATION SCALED AT PI/2 R/S2
38 #
39 QRNDXER EQUALS ITEMP6
40 OCT23146 OCTAL 23146 # DECIMAL .6
41 NZACCDOT EQUALS ITEMP3
42 #
43 TIMEGMBL CAF ONE # INITIALZE ALLOWGTS.
44 TS ALLOWGTS
45 #
46 CAF TWO # SET UP LOOP FOR R AXIS.
47 LXCH Q # SAVE RETURN ADDRESS.
48 LXCH RUPTREG2
49
50
51
52
53
54
55
56
57
58
59
60
```



```
1 TIMQGBL TCF +2
2 CAF ZERO # NOW DO THE Q-AXIS
3
4 TS QRNDXER
5 INDEX QRNDXER
6 CA ACCDOTQ # ACCDOT IS PRESUMED TO BE AT PI/2(7).
7
8 EXTEND
9 BZMF TGOFFNOW # IS ACCDOT LESS THAN OR EQUAL TO 0?
10 TS NZACCDOT # NO. STORE NON-ZERO, POSITIVE ACCDOT.
11
12 ALPHATRY INDEX QRNDXER
13 CS AOSQ
14
15 EXTEND
16 BZF TGOFFNOW # IS ALPHA ZERO?
17
18 TS Q # SAVE A COPY OF -AOS.
19 EXTEND # NO. RESCALE FOR TIMEGBL USE.
20 MP OCT23146 # OCTAL 23146 IS DECIMAL .6
21
22 AD Q # -1.6*AOS AT PI/2 = -.4*AOS AT PI/8.
23 TS L # WAS THERE OVERFLOW?
24 TCF SETNEGU # NO. COMPUTE DRIVE TIME.
25
26 CS A # RECOVER -SGN(AOS) IN THE A REGISTER.
27 INDEX QRNDXER # YES. START DRIVE WITHOUT WAITLIST.
28
29 XCH NEGUQ
30 TCF NOTALLOW # KNOCK DOWN THE ALLOWGTS FLAG.
31
32 SETNEGU EXTEND
33 BZMF POSALPH
34
35 COM
36 TS ITEMP2 # STORE -ABS(.4*AOS) SCALED AT PI/8.
37 CS BIT1
38
39 TCF POSALPH +2
40 TS ITEMP2 # STORE -ABS(.4*AOS) SCALED AT PI/8.
41 CA BIT1
42
43 +2 INDEX QRNDXER # SGN(AOS) INTO NEGU
44 TS NEGUQ # STORE SGN(ALPHA) AS NEGU
45
46 CA NZACCDOT
47 EXTEND
48 MP BIT12 # 2*ACCDOT, SCALED AT PI/8.
49 AD ITEMP2 # -ABS(ALPHS) + 2*ACCDOT, AT PI/8.
50
51 EXTEND
52 BZMF NOTALLOW # IS DRIVE TIME MORE THAN TWO SECONDS?
53 CS ITEMP2 # NO. COMPUTE DRIVE TIME.
54 EXTEND # ABS(ALPHA) AT PI/8.
55 MP OCT00240 # DECIMAL 10/1024
56
57 EXTEND # QUOTIENT IS DRIVE TIME AT WAITLIST.
58 DV NZACCDOT # ABS(ALPHA)/ACCDOT AT 2(14)/100
59
60
```

|    |          |        |          |                                           |    |
|----|----------|--------|----------|-------------------------------------------|----|
| 1  |          |        |          |                                           | 1  |
| 2  |          | EXTEND |          |                                           | 2  |
| 3  |          | BZF    | TGOFFNOW | # DRIVE TIME MUST BE GREATER THAN ZERO.   | 3  |
| 4  |          |        |          |                                           | 4  |
| 5  |          | TCF    | DRIVEON  |                                           | 5  |
| 6  |          |        |          |                                           | 6  |
| 7  | TGOFFNOW | CAF    | ZERO     | # TURN OFF GIMBAL NOW.                    | 7  |
| 8  |          | INDEX  | QRNDXER  |                                           | 8  |
| 9  |          | TS     | NEGUQ    |                                           | 9  |
| 10 |          |        |          |                                           | 10 |
| 11 |          | TCF    | DONEYET  |                                           | 11 |
| 12 |          |        |          |                                           | 12 |
| 13 | NOTALLOW | CAF    | OCT31    |                                           | 13 |
| 14 |          | INDEX  | QRNDXER  |                                           | 14 |
| 15 |          | TS     | QGIMTIMR |                                           | 15 |
| 16 |          | CAF    | ZERO     | # DRIVE TIME IS MORE THAN 2 SECONDS, SO   | 16 |
| 17 |          | TS     | ALLOWGTS | # DO NOT PERMIT FURTHER GTS ATTITUDE-RATE | 17 |
| 18 |          |        |          | # CONTROL UNTIL AOSTASK APPROVES.         | 18 |
| 19 |          | TCF    | DONEYET  | # NO WAITLIST CALL IS MADE.               | 19 |
| 20 |          |        |          |                                           | 20 |
| 21 | DRIVEON  | INDEX  | QRNDXER  |                                           | 21 |
| 22 |          | TS     | QGIMTIMR | # CHOOSE Q OR R AXIS.                     | 22 |
| 23 |          |        |          |                                           | 23 |
| 24 | DONEYET  | CCS    | QRNDXER  |                                           | 24 |
| 25 |          | TCF    | TIMQGMBL |                                           | 25 |
| 26 |          |        |          |                                           | 26 |
| 27 |          | DXCH   | RUPTREG3 | # PROTECT IBNKCALL ERASABLES. ACDT+C12    | 27 |
| 28 |          | DXCH   | ITEMP2   | # LEAVES ITEMS2,3 ALONE.                  | 28 |
| 29 |          |        |          |                                           | 29 |
| 30 |          | TC     | IBNKCALL | # TURN OFF CHANNEL BITS, SET Q(R)ACCDOTS. | 30 |
| 31 |          | CADR   | ACDT+C12 |                                           | 31 |
| 32 |          |        |          |                                           | 32 |
| 33 |          | DXCH   | ITEMP2   | # RESTORE ERASABLES FOR IBNKCALL.         | 33 |
| 34 |          | DXCH   | RUPTREG3 |                                           | 34 |
| 35 |          |        |          |                                           | 35 |
| 36 |          | TC     | RUPTREG2 | # RETURN TO CALLER.                       | 36 |
| 37 |          |        |          |                                           | 37 |
| 38 | OCT00240 | OCTAL  | 00240    | # DECIMAL 10/1024                         | 38 |
| 39 |          |        |          |                                           | 39 |
| 40 |          |        |          |                                           | 40 |
| 41 |          |        |          |                                           | 41 |
| 42 |          |        |          |                                           | 42 |
| 43 |          |        |          |                                           | 43 |
| 44 |          |        |          |                                           | 44 |
| 45 |          |        |          |                                           | 45 |
| 46 |          |        |          |                                           | 46 |
| 47 |          |        |          |                                           | 47 |
| 48 |          |        |          |                                           | 48 |
| 49 |          |        |          |                                           | 49 |
| 50 |          |        |          |                                           | 50 |
| 51 |          |        |          |                                           | 51 |
| 52 |          |        |          |                                           | 52 |
| 53 |          |        |          |                                           | 53 |
| 54 |          |        |          |                                           | 54 |
| 55 |          |        |          |                                           | 55 |
| 56 |          |        |          |                                           | 56 |
| 57 |          |        |          |                                           | 57 |
| 58 |          |        |          |                                           | 58 |
| 59 |          |        |          |                                           | 59 |
| 60 |          |        |          |                                           | 60 |

# THE FOLLOWING SECTION IS A CONTINUATION OF THE TRIM GIMBAL CONTROL FROM THE LAST GTS ENTRY. THE QUANTITY NEGUSUM  
# IS COMPUTED FOR EACH AXIS (Q,R),  $.707*DEL*FUNCTION(3/2) + K2THETA = NEGUSUM$ . NEW DRIVES ARE ENTERED TO CH 12.  
#  
# THE SUBROUTINE GTSQRT ACCEPTS A DOUBLE PRECISION VALUE IN FUNCTION, FUNCTION +1 AND RETURNS A SINGLE-PRECISION  
# SQUARE ROOT OF THE FOURTEEN MOST SIGNIFICANT BITS OF THE ARGUMENT. ALSO, THE CELL SHFTFLAG CONTAINS A BINARY  
# EXPONENT S, SUCH THAT THE SQUARE ROOT (RETURNED IN THE A REGISTER) MUST BE SHIFTED RIGHT (MULTIPLIED BY 2 TO THE  
# POWER (-S)) IN ORDER TO BE THE TRUE SQUARE ROOT OF THE FOURTEEN MOST SIGNIFICANT BITS OF FUNCTION, FUNCTION +1.  
# SQUARE ROOT ERROR IS NOT MORE THAN 2 IN THE 14TH SIGNIFICANT BIT. CELLS CLOBBED ARE A,L,SHFTFLAG,ININDEX,  
# HALFARG,SCRATCH,SR,FUNCTION, FUNCTION +1. GTSQRT IS CALLED BY TC GTSQRT AND RETURNS VIA TC Q OR TC FUNCTION +1.  
# ZERO OR NEGATIVE ARGUMENTS YIELD ZERO FOR SQUARE ROOTS.

|        |     |          |                                           |
|--------|-----|----------|-------------------------------------------|
| GTSQRT | CCS | FUNCTION |                                           |
|        | TCF | GOODARG  | # FUNCTION IS POSITIVE. TAKE SQUARE ROOT. |
|        | TCF | +2       | # HIGH ORDER WORD IS ZERO. TRY THE LOWER. |
|        | TCF | ZERROOT  | # NEGATIVE. USE ZERO FOR 1/2 POWER.       |

|    |             |
|----|-------------|
| CA | FUNCTION +1 |
|----|-------------|

|        |         |
|--------|---------|
| EXTEND |         |
| BZMF   | ZERROOT |

|         |     |          |            |
|---------|-----|----------|------------|
| ZERROOT | TCF | ZEROHIGH | # PROCEED. |
|         | CA  | ZERO     |            |
|         | TS  | SHFTFLAG |            |
|         | TC  | Q        |            |

|          |     |             |                                       |
|----------|-----|-------------|---------------------------------------|
| ZEROHIGH | XCH | FUNCTION    | # 14 MOST SIGNIFICANT BITS ARE IN THE |
|          | XCH | FUNCTION +1 | # LOWER WORD. EXCHANGE THEM.          |
|          | CA  | SEVEN       |                                       |
|          | TCF | GOODARG +1  |                                       |

|         |     |          |                                |
|---------|-----|----------|--------------------------------|
| GOODARG | CA  | ZERO     |                                |
|         | TS  | SHFTFLAG |                                |
|         | CA  | TWELVE   | # INITIALIZE THE SCALING LOOP. |
|         | TS  | ININDEX  |                                |
|         | TCF | SCALLOOP |                                |

|          |     |          |  |
|----------|-----|----------|--|
| SCALSTRT | CA  | FUNCTION |  |
|          | TCF | SCALDONE |  |

|         |        |          |                                          |
|---------|--------|----------|------------------------------------------|
| MULBUSH | CA     | NEG2     | # IF ARG IS NOT LESS THAN 1/4, INDEX IS  |
|         | ADS    | ININDEX  | # ZERO, INDICATING NO SHIFT NEEDED.      |
|         | EXTEND |          | # BRANCH IF ARG IS NOT LESS THAN 1/4.    |
|         | BZMF   | SCALSTRT | # OTHERWISE COMPARE ARG WITH A REFERENCE |
|         |        |          | # WHICH IS 4 TIMES LARGER THAN THE LAST. |

|          |        |          |                                            |
|----------|--------|----------|--------------------------------------------|
| SCALLOOP | CS     | FUNCTION |                                            |
|          | INDEX  | ININDEX  |                                            |
|          | AD     | BIT15    | # REFERENCE MAGNITUDE LESS OR EQUAL TO 1/4 |
|          | EXTEND |          |                                            |
|          | BZMF   | MULBUSH  | # IF ARG IS NOT LESS THAN REFERENCE, GO    |
|          |        |          | # AROUND THE MULBERRY BUSH ONCE MORE.      |

```
1
2 INDEX ININDEX
3 CA BIT15 # THIS IS THE SCALE MAGNITUDE
4 XCH HALFARG # 2**(-ININDEX) IS THE SHIFT DIVISOR.
5 EXTEND
6 DCA FUNCTION # RESCALE ARGUMENT.
7 EXTEND
8 DV HALFARG
9
10 # ININDEX AND SHFTFLAG PRESERVE INFO FOR
11 # RESCALING AFTER ROOT PROCESS.
12
13 SCALDONE EXTEND
14 QXCH FUNCTION +1 # SAVE Q FOR RETURN
15
16 EXTEND
17 MP BIT14
18 TS HALFARG
19
20 MASK BIT13
21 CCS A
22 CA OCT11276
23
24 AD ROTHALF # INITIAL GUESS IS ROOT 1/2 OR POSMAX
25 TC ROOTCYCL
26 TC ROOTCYCL
27
28 TC ROOTCYCL
29 TC FUNCTION +1
30
31 # *****
32
33
34 RSTOFGTS TC GTSQRT
35 PRODUCT XCH K2CNTRAL
36
37 EXTEND
38 MP K2CNTRAL
39 DXCH K2CNTRAL
40
41 EXTEND
42 MP L # THE PRODUCT OF
43 ADS K2CNTRAL +1 # 1/2 2 1/2
44 # K *(DEL*OMEGA + ALPHA /(2*K))
45
46 TS L # AND
47 TCF +2 #
48 ADS K2CNTRAL # DEL*(DEL*OMEGA + ALPHA /(2*K)) NOW IN
49 # K2CNTRAL
50
51
52 DOSHIFT CA ININDEX
53
54 EXTEND # MULTIPLY IN THE FACTOR 2(-S), RETURNED
55 MP # BY THE GTSQRT SUBROUTINE
56 ADS BIT14
57 SHFTFLAG
58
59 EXTEND
60 BZF ADDITIN
61 INDEX SHFTFLAG
62 CA BIT15
```

|          |        |             |                                           |
|----------|--------|-------------|-------------------------------------------|
|          | XCH    | K2CNTRAL    |                                           |
|          | EXTEND |             |                                           |
|          | MP     | K2CNTRAL    |                                           |
|          | DAS    | K2THETA     |                                           |
|          | XCH    | K2CNTRAL    |                                           |
|          | EXTEND |             |                                           |
|          | MP     | K2CNTRAL +1 |                                           |
|          | ADS    | K2THETA +1  |                                           |
|          | TS     | L           |                                           |
|          | TCF    | +2          |                                           |
|          | ADS    | K2THETA     |                                           |
|          | TCF    | NEGUSUM     |                                           |
| ADDITIN  | EXTEND |             |                                           |
|          | DCA    | K2CNTRAL    |                                           |
|          | DAS    | K2THETA     | # NOW ADD IN THE K2THETA TERM.            |
| NEGUSUM  | CCS    | K2THETA     | # TEST SIGN OF HIGH ORDER PART.           |
|          | TCF    | NEGDRIVE    |                                           |
|          | TCF    | +2          |                                           |
|          | TCF    | POSDRIVE    |                                           |
|          | CCS    | K2THETA +1  | # SIGN TEST FOR LOW ORDER PART.           |
| NEGDRIVE | CA     | BIT1        |                                           |
|          | TCF    | +2          | # STOP GIMBAL DRIVE FOR A ZERO NEGUSUM.   |
| POSDRIVE | CS     | BIT1        |                                           |
|          | TS     | L           | # SAVE FOR DRIVE REVERSAL TEST.           |
|          | INDEX  | QRCNTR      |                                           |
|          | XCH    | NEGUQ       |                                           |
|          | EXTEND |             |                                           |
|          | MP     | L           | # MULTIPLY OLD NEGU AND NEW NEGU.         |
|          | CCS    | L           |                                           |
|          | TCF    | LOUPE       | # NON-ZERO GIMBAL DRIVE BEING CONTINUED.  |
|          | TCF    | ZEROLOUP    | # NO REVERSAL PROBLEM HERE.               |
|          | TCF    | REVERSAL    | # NON-ZERO GIMBAL DRIVE BEING REVERSED.   |
|          | TCF    | ZEROLOUP    | # NO REVERSAL PROBLEM HERE.               |
| REVERSAL | INDEX  | QRCNTR      | # A ZERO-DRIVE PAUSE IS NEEDED HERE. ZERO |
|          | TS     | QACCDOT     | # IS IN A REGISTER FROM CCS ON (-1).      |
|          | INDEX  | QRCNTR      |                                           |
|          | CS     | GMBLBITA    |                                           |
|          | EXTEND |             |                                           |
|          | WAND   | CHAN12      |                                           |
| ZEROLOUP | CS     | RCSFLAGS    | # SET UP REQUEST FOR ACDT+C12 CALL.       |
|          | MASK   | CALLGMBL    |                                           |
|          | ADS    | RCSFLAGS    |                                           |

```
1 LOUPE CCS QRCNTR # HAVE BOTH AXES BEEN PROCESSED?
2 TCF GOQTRIMG # NO. DO Q AXIS NEXT.
3
4
5 CA SAVESR # RESTORE THE SR
6 TS SR
7
8 GOCLOSE EXTEND # TERMINATE THE JASK.
9 DCA CLOSEADR
10 DTCB
11
12 EBANK= AOSQ
13 CLOSEADR 2CADR CLOSEOUT # TERMINATE THE JASK.
14
15 TWELVE EQUALS OCT14
16 ROTHALF OCTAL 26501 # SQUARE ROOT OF 1/2
17 GMBLBITA OCTAL 01400 # INDEXED WRT GMBLBITB DO NOT MOVE*****
18 OCT11276 OCTAL 11276 # POSMAX - ROTHALF
19 GMBLBITB OCTAL 06000 # INDEXED WRT GMBLBITA DO NOT MOVE*****
```

```
21 # SUBROUTINE ROOTCYCL: BY CRAIG WORK,3 APRIL 68
```

```
22 # ROOTCYCL IS A SUBROUTINE WHICH EXECUTES ONE NEWTON SQUARE ROOT ALGORITHM ITERATION. THE INITIAL GUESS AT THE
23 # SQUARE ROOT IS PRESUMED TO BE IN THE A REGISTER AND ONE-HALF THE SQUARE IS TAKEN FROM HALFARG. THE NEW APPROXI-
24 # MATION TO THE SQUARE ROOT IS RETURNED IN THE A REGISTER. DEBRIS: A,L,SR,SCRATCH. ROOTCYCL IS CALLED FROM
25 # LOCATION (LOC) BY A TC ROOTCYCL, AND RETURNS (TC Q) TO LOC +1.
26 # WARNING: IF THE INITIAL GUESS IS NOT GREATER THAN THE SQUARE, DIVIDE OR ADD OVERFLOW IS A REAL POSSIBILITY.
```

```
28 ROOTCYCL TS SCRATCH # STORE X
29 TS SR # X/2 NOW IN SR
30 CA HALFARG # ARG/2 IN THE A REG
31 ZL
32 EXTEND
33 DV SCRATCH # (ARG/X)/2
34 AD SR # (X + ARG/X)/2 IN THE A REG
35 TC Q
```

```
1 # PROGRAM NAME: 1/ACCS
2 # PROGRAM WRITTEN BY: BOB COVELLI AND MIKE HOUSTON
3 # LAST MODIFICATION: FEB. 14, 1969 BY G. KALAN
4 #
5 # PROGRAM DESCRIPTION:
6 # 1/ACCS PROVIDES THE INTERFACE BETWEEN THE GUIDANCE PROGRAMS AND THE DIGITAL AUTOPILOT. WHENEVER THERE IS A
7 # CHANGE IN THE MASS OF THE VEHICLE, IN THE DEADBAND SELECTED, IN THE VEHICLE CONFIGURATION (ASCENT-DESCENT-
8 # DOCKED), AND DURING A FRESH START OR A RESTART, 1/ACCS IS CALLED TO COMMUNICATE THE DATA CHANGES TO THE DAP.
9 #
10 #
11 # THE INPUTS TO 1/ACCS ARE MASS, ACCELERATION (ABDELV), DEADBAND (DB), OFFSET ACCELERATIONS (AOSQ AND AOSR),
12 # STAGE VERIFY BIT (CHAN30, BIT2), DOCKED BIT (DAPBOOLS, BIT13), DRIFT BIT (DAPBOOLS, BIT8), USEQRJTS (DAPBOOLS,
13 # BIT14), AND SURFACE FLAG (FLAGWRDB, BIT8), AND CH5MASK.
14 #
15 # 1/ACCS COMPUTES THE JET ACCELERATIONS (1JACC, 1JACCQ, 1JACCR) AS FUNCTIONS OF MASS. 1JACCU AND 1JACCV ARE
16 # FORMED BY RESOLVING 1JACCQ AND 1JACCR. IN THE DESCENT CASE, THE DESCENT ENGINE MOMENT ARM (L, PVT-CG) IS ALSO
17 # COMPUTED AS A FUNCTION OF MASS. THE RATE OF CHANGE OF ACCELERATION DUE TO ROTATION OF THE GIMBAL (ACCDOTQ,
18 # ACCDOTR) IS ALSO COMPUTED IN THE DESCENT CASE.
19 #
20 # AFTER THE ABOVE COMPUTATIONS, THE PROGRAM 1/ACCONT COMPUTES THE RECIPROCAL NET ACCELERATIONS ABOUT THE P, U,
21 # AND V AXES (2 JETS FOR P-AXIS, BOTH 1 AND 2 JETS FOR U AND V AXES), AND THE RECIPROCAL COAST ACCELERATIONS ABOUT
22 # THE P, U, AND V AXES. THE ACCELERATION FUNCTIONS (ACCFCTZ1 AND ACCFCTZ5) ARE ALSO COMPUTED FOR THESE AXES. THE
23 # FIRE AND COAST DEADBANDS AND AXISDIST ARE COMPUTED FOR EACH AXIS. FLAT AND ZONE3LIM, THE WIDTH AND HEIGHT OF THE
24 # MINIMUM IMPULSE ZONE, ARE COMPUTED. 1/ACCONT ALSO SETS ACCSWU AND ACCSWV, WHICH INDICATE WHEN 1 JET ACCELERATION
25 # IS NOT SUFFICIENT TO PRODUCE MINIMUM ACCELERATION. AT THE COMPLETION OF 1/ACCS, THE ACCSOKAY BIT IS SET.
26 #
27 # SUBROUTINES CALLED:
28 # TIMEGMBL
29 # MAKECADR
30 # ROT45DEG
31 #
32 # CALLING SEQUENCE:
33 # TC BANKCALL # (1/ACCS MUST BE CALLED BY BANKCALL)
34 # CADR 1/ACCS
35 #
36 # NORMAL EXIT: VIA BANKJUMP
37 #
38 # ALARM AND EXIT MODES: NONE
39 #
40 # INPUT/OUTPUT: SEE PROGRAM DESCRIPTION.
41 #
42 # DEBRIS:
43 # ALL OF THE EXECUTIVE TEMPORARY REGISTERS, EXCEPT FIXLOC AND OVFind, AND THE CORE SET AREA FROM MPAC TO BANKSET.
44 #
45 # RESTRICTIONS:
46 # 1/ACCS MUST BE CALLED BY BANKCALL
47 # EBANK IS SET TO 6, BUT NOT RESTORED.
```

```
1
2
3 BANK 20
4 SETLOC DAPS3
5 BANK
6
7 COUNT* $$/DAPAD
8
9 EBANK= AOSQ
10
11 # ENTRY IS THROUGH 1/ACCJOB OR 1/ACCSET WHEN 1/ACCS IS TO BE DONE AS A SEPARATE NOVAC JOB.
12 # IT IS POSSIBLE FOR MORE THAN ONE OF THESE JOBS TO BE SET UP CONCURRENTLY. HOWEVER, SINCE THERE IS NO CHECK OF
13 # NEWJOB, A SECOND MANIFESTATION CANNOT BE STARTED UNTIL THE FIRST IS COMPLETED.
14
15 1/ACCSET CAF ZERO # ENTRY FROM FRESH START/RESTART CODING.
16 TS AOSQ # NULL THE OFFSET ESTIMATES FOR 1/ACCS.
17 TS AOSR
18 TS ALPHAQ # NULL THE OFFSET ESTIMATES FOR DOWNLIST
19 TS ALPHAR
20
21 1/ACCJOB TC BANKCALL # 1/ACCS ASSUMES ENTRY VIA BANKCALL.
22 CADR 1/ACCS +2 # SKIP EBANK SETTING.
23
24 TC ENDOFJOB
25
26 1/ACCS CA EBANK6 # ***** EBANK SET BUT NOT RESTORED *****
27 TS EBANK
28
29 TC MAKECADR # SAVE RETURN SO THAT BUF2 MAY BE USED
30 TS ACCRETRN
31
32 # DETERMINE MASS OF THE LEM.
33
34 CA DAPBOOLS # IS THE CSM DOCKED
35 MASK CSMDOCKD
36 TS DOCKTEMP # STORE RECORD OF STATE IN TEMP (MPAC +3).
37
38 CCS A
39 CS CSMMASS # DOCKED: LEMMAS = MASS - CSMMASS
40 AD MASS # LEM ALONE: LEMMASS = MASS
41 TS LEMMASS
42
43 # ON THE BASIS OF APSFLAG:
44 # SET THE P-AXIS RATE COMMAND LIMIT FOR 2-JET/2-JET CONTROL
45 # SET MPAC, WHICH INDICATES THE PROPER SET OF COEFFICIENTS FOR THE LEM-ALONE F(MASS) CALCULATIONS
46 # ENSURE THAT THE LEM MASS VALUE IS WITHIN THE ACCEPTABLE RANGE
47
48 INHINT
49 CAE FLGWRD10 # DETERMINE WHETHER STAGED.
50 MASK APSFLBIT
51 EXTEND
52 BZF DPSFLITE
```



```
1
2 CS POSMAX # ASCENT (OR ON LUNAR SURFACE)
3 TS -2JETLIM # ALWAYS 2 JETS FOR P-AXIS RATE COMMAND
4 CAF OCT14 # INITIALIZE INDEX AT 12.
5 TS MPAC
6 CS LEMMASS # CHECK IF MASS TOO HIGH. CATCH STAGING.
7 AD HIASCENT
8 EXTEND
9 BZMF MASSFIX
10 CS LEMMASS # CHECK IF MASS TOO LOW. THIS LIMITS THE
11 AD LOASCENT # DECREMENTING BY MASSMON.
12 EXTEND
13 BZMF F(MASS)
14
15 MASSFIX ADS LEMMASS # STORE THE VIOLATED LIMIT AS LEMMASS.
16 ZL
17 CCS DOCKTEMP # ALSO CORRECT TOTAL MASS, ZEROING THE
18 CAE CSMMASS # LOW-ORDER WORD.
19 AD LEMMASS # DOCKED: MASS = LEMMASS + CSMMASS
20 DXCH MASS
21 TCF F(MASS) # LEM ALONE: MASS = LEMMASS
22
23 DPSFLITE CS BIT10 # FOUR JETS FOR P-AXIS RATE COMMAND ERRORS
24 TS -2JETLIM # EXCEEDING 1.4 DEG/SEC (SCALED AT 45)
25 CAF SIX
26 TS MPAC
27 CS LEMMASS # CHECK IF MASS TOO HIGH. SHOULD NEVER
28 AD HIDESCNT # OCCUR EXCEPT PERHAPS BEFORE THE PAD
29 EXTEND # LOAD IS DONE.
30 BZMF MASSFIX
31 CS LEMMASS # CHECK IF MASS TOO LOW. THIS LIMITS THE
32 AD LODESCNT # DECREMENTING BY MASSMON.
33 AD HIASCENT
34 EXTEND
35 BZMF F(MASS)
36 TCF MASSFIX
37
38 # COMPUTATION OF FUNCTIONS OF MASS
39
40 F(MASS) RELINT
41 CCS DOCKTEMP
42 TCF DOCKED # DOCKED: USE SEPARATE COMPUTATION.
43
44 STCTR CA TWO
45 TS MPAC +1 # J=2,1,0 FOR 1JACCR,1JACCQ,1JACC
46
47 CS TWO
48 ADS MPAC # JX=10,8,6 OR 4,2,0 TO INDEX COEFS.
49
50 STCTR1 CAE LEMMASS
51 INDEX MPAC
52 AD INERCONC
53 TS MPAC +2 # MASS + C
```

```
1
2 EXTEND
3 INDEX MPAC
4 DCA INERCONA
5 EXTEND
6 DV MPAC +2
7 INDEX MPAC
8 AD INERCONB
9 INDEX MPAC +1 # 1JACC(J)=A(JX)/(MASS+C(JX) + B(JX)
10 TS 1JACC # 1JACC(-1)=L,PVT-CG SCALED AT 8 FEET
11
12 CCS MPAC +1
13 TCF STCTR
14 TCF COMMEQS
15 TCF LRESC
16
17 # COEFFQ AND COEFFR ARE COMPUTED IN THIS SECTION. THEY ARE USED TO RESOLVE Q-R COMPONENTS INTO NON-ORTHOGONAL
18 # U AND V COMPONENTS (SEE ROT-TOUV SECTION).
19
20 COMMEQS CS 1JACCR
21 AD 1JACCQ
22
23 EXTEND
24 BZMF BIGIQ
25 EXTEND # EPSILON IS A MEASURE OF COUPLING AND IS
26 DV 1JACCQ # DEFINED=1-IQ/IR FOR IR GREATER THAN IQ.
27 TS EPSILON # THE COMPUTED EXPRESSION IS EQUIVALENT
28 AD -EPSMAX
29
30 EXTEND
31 BZMF GOODEPS1
32 CS -EPSMAX
33 TS EPSILON # EPSILON IS LIMITED TO A MAX. OF .42265
34 GOODEPS1 CA EPSILON
35 EXTEND
36 MP 0.35356
37 AD .7071
38 TS COEFFR # IN THIS CASE WHERE IR IS GREATER THAN
39 CS POSMAX # IQ, COEFFQ=-.707(1+.5EPSILON)(1-EPSILON)
40 AD EPSILON # AND COEFFR=.707(1+.5EPSILON)
41 EXTEND
42 MP COEFFR
43 TS COEFFQ
44 TCF JACCUV
45 BIGIQ EXTEND # EPSILON IS DEFINED AS 1-IR/IQ FOR IQ
46 DV 1JACCR # GREATER THAN IR. -EPSILON IS COMPUTED
47 TS -EPSILON # RATHER THAN EPSILON FOR CONVENIENCE
48 CS -EPSILON
49 AD -EPSMAX
50 EXTEND
51 BZMF GOODEPS2
52 CA -EPSMAX
53 TS -EPSILON # EPSILON IS LIMITED TO A MAX. OF .42265
54
55
56
57
58
59
60
```

```
1 GOODEPS2 CA -EPSILON
2
3 EXTEND
4 MP 0.35356
5 AD -.7071
6 TS COEFFQ # IN THIS CASE WHERE IQ IS GREATER THAN
7 CS -EPSILON # IR, COEFFQ=-.707(1+.5EPSILON) AND
8 AD NEGMAX # COEFFR=.707(1+.5EPSILON)(1-EPSILON)
9 EXTEND
```

```
10 JACCUV MP COEFFQ
11 TS COEFFR
12 CS COEFFQ
13 EXTEND
14 MP 1JACCQ # 1JACCQ IS SCALED AT PI/4
15 TS 1JACCU # 1JACCU USED AS TEMPORARY STORAGE
```

```
16 CA COEFFR
17 EXTEND
18 MP 1JACCR
19 AD 1JACCU
20 EXTEND
21 MP BIT14 # SCALING CHANGED FROM PI/4 TO PI/2
22 TS 1JACCU
23 TS 1JACCV # SCALED AT PI/2 RADIANS/SEC(2)
24 CCS MPAC # COMPUTE L,PVT-CG IF IN DESCENT
25 CAF ZERO # ZERO SWITCHES AND GO TO 1/ACCONT IN
26 TS ALLOWGTS # ASCENT
27 TCF 1/ACCONT -1
```

```
28
29 CS TWO
30 TS MPAC
31 CS ONE
32 TS MPAC +1
33 TCF STCTR1
```

```
34
35 # THIS SECTION COMPUTES THE RATE OF CHANGE OF ACCELERATION DUE TO THE ROTATION OF THE GIMBALS. THE EQUATION
36 # IMPLEMENTED IN BOTH THE Y-X PLANE AND THE Z-X PLANE IS -- D(ALPHA)/DT = TL/I*D(DELTA)/DT, WHERE
37 # T = ENGINE THRUST FORCE
38 # L = PIVOT TO CG DISTANCE OF ENGINE
39 # I = MOMENT OF INERTIA
```

```
40
41 LRESC CAE ABDELV # SCALED AT 2(13) CM/SEC(2)
42 EXTEND
43 MP MASS # SCALED AT B+16 KGS
44 TC DVOVSUB # GET QUOTIENT WITH OVERFLOW PROTECTION
45 ADRES GFACTM
```

```
46
47 # MASS IS DIVIDED BY ACCELERATION OF GRAVITY IN ORDER TO MATCH THE UNITS OF IXX,IYY,IZZ, WHICH ARE SLUG-FT(2).
48 # THE RATIO OF ACCELERATION FROM PIPAS TO ACCELERATION OF GRAVITY IS THE SAME IN METRIC OR ENGINEERING UNITS, SO
49 # THAT IS UNCONVERTED. 2.20462 CONVERTS KG. TO LB. NOW T IN IN A SCALED AT 2(14).
```

```
50
51 EXTEND
52 MP L,PVT-CG # SCALED AT 8 FEET.
```

```
1
2 INHINT
3 TS MPAC
4 EXTEND
5 MP 1JACCR
6 TC DVOVSUB # GET QUOTIENT WITH OVERFLOW PROTECTION
7 ADRES TORKJET1
8
9 TS ACCDOTR # SCALED AT PI/2(7)
10 CA MPAC
11 EXTEND
12 MP 1JACCQ
13 TC DVOVSUB # GET QUOTIENT WITH OVERFLOW PROTECTION
14 ADRES TORKJET1
15
16 SPSCONT TS ACCDOTQ # SCALED AT PI/2(7)
17 EXTEND
18 MP DGBF # .3ACCDOTQ SCALED AT PI/2(8)
19 TS KQ
20 CAE ACCDOTR # .3ACCDOTR AT PI/2(8)
21 EXTEND
22 MP DGBF
23 TS KRDP
24 EXTEND # NOW COMPUTE QACCDOT, RACCDOT, THE SIGNED
25 READ CHAN12 # JERK TERMS. STORE CHANNEL 12. WITH GIMBAL
26 TS MPAC +1 # DRIVE BITS 9 THROUGH 12 SET LOOP
27 CAF BIT2 # INDEX TO COMPUTE RACCDOT, THEN QACCDOT.
28 TCF LOOP3
29 CAF ZERO # ACCDOTQ AND ACCDOTR ARE NOT NEGATIVE,
30 TS MPAC # BECAUSE THEY ARE MAGNITUDES
31 LOOP3 CA MPAC +1
32 INDEX MPAC
33 MASK GIMBLBTS # MASK CHANNEL IMAGE FOR ANY GIMBAL MOTION
34 EXTEND
35 BZF ZACCDOT # IF NONE, Q(R)ACCDOT IS ZERO.
36 CA MPAC +1
37 INDEX MPAC # GIMBAL IS MOVING. IS ROTATION POSITIVE.
38 MASK GIMBLBTS +1
39 EXTEND
40 BZF FRSTZERO # IF NOT POSITIVE, BRANCH
41 INDEX MPAC # POSITIVE ROTATION, NEGATIVE Q(R)ACCDOT.
42 CS ACCDOTQ
43 TCF STACCDOT
44 INDEX MPAC # NEGATIVE ROTATION, POSITIVE Q(R)ACCDOT.
45 CA ACCDOTQ
46 TCF STACCDOT
47 CAF ZERO
48 INDEX MPAC
49 TS QACCDOT # STORE Q(R)ACCDOT.
50 CCS MPAC
51 TCF LOOP3 -1 # NOW DO QACCDOT.
```

|    |          |        |          |                                            |    |
|----|----------|--------|----------|--------------------------------------------|----|
| 1  |          |        |          |                                            | 1  |
| 2  |          | CS     | DAPBOOLS | # IS GIMBAL USABLE?                        | 2  |
| 3  |          | MASK   | USEQRJTS |                                            | 3  |
| 4  |          | EXTEND |          |                                            | 4  |
| 5  |          | BZF    | DOWNGTS  | # NO. BE SURE THE GIMBAL SWITCHES ARE DOWN | 5  |
| 6  |          | CS     | T5ADR    | # YES. IS THE DAP RUNNINT?                 | 6  |
| 7  |          | AD     | PAXISADR |                                            | 7  |
| 8  |          | EXTEND |          |                                            | 8  |
| 9  |          | BZF    | +2       |                                            | 9  |
| 10 |          | TCF    | DOWNGTS  | # NO. BE SURE THE GIMBAL SWITCHES ARE DOWN | 10 |
| 11 |          | CCS    | INGTS    | # YES. IS GTS IN CONTROL?                  | 11 |
| 12 |          | TCF    | DOCKTEST | # YES. PROCEED WITH 1/ACCS.                | 12 |
| 13 |          | TC     | IBNKCALL | # NO. NULL OFFSET AND FIND ALLOWGTS        | 13 |
| 14 |          | CADR   | TIMEGMBL |                                            | 14 |
| 15 |          |        |          |                                            | 15 |
| 16 | DOCKTEST | CCS    | DOCKTEMP | # BYPASS 1/ACCONT WHEN DOCKED.             | 16 |
| 17 |          | TCF    | 1/ACCRET |                                            | 17 |
| 18 |          | TCF    | 1/ACCONT |                                            | 18 |
| 19 |          |        |          |                                            | 19 |
| 20 |          |        |          |                                            | 20 |
| 21 |          |        |          |                                            | 21 |
| 22 |          |        |          |                                            | 22 |
| 23 |          |        |          |                                            | 23 |
| 24 |          |        |          |                                            | 24 |
| 25 |          |        |          |                                            | 25 |
| 26 |          |        |          |                                            | 26 |
| 27 |          |        |          |                                            | 27 |
| 28 |          |        |          |                                            | 28 |
| 29 |          |        |          |                                            | 29 |
| 30 |          |        |          |                                            | 30 |
| 31 |          |        |          |                                            | 31 |
| 32 |          |        |          |                                            | 32 |
| 33 |          |        |          |                                            | 33 |
| 34 |          |        |          |                                            | 34 |
| 35 |          |        |          |                                            | 35 |
| 36 |          |        |          |                                            | 36 |
| 37 |          |        |          |                                            | 37 |
| 38 |          |        |          |                                            | 38 |
| 39 |          |        |          |                                            | 39 |
| 40 |          |        |          |                                            | 40 |
| 41 |          |        |          |                                            | 41 |
| 42 |          |        |          |                                            | 42 |
| 43 |          |        |          |                                            | 43 |
| 44 |          |        |          |                                            | 44 |
| 45 |          |        |          |                                            | 45 |
| 46 |          |        |          |                                            | 46 |
| 47 |          |        |          |                                            | 47 |
| 48 |          |        |          |                                            | 48 |
| 49 |          |        |          |                                            | 49 |
| 50 |          |        |          |                                            | 50 |
| 51 |          |        |          |                                            | 51 |
| 52 |          |        |          |                                            | 52 |
| 53 |          |        |          |                                            | 53 |
| 54 |          |        |          |                                            | 54 |
| 55 |          |        |          |                                            | 55 |
| 56 |          |        |          |                                            | 56 |
| 57 |          |        |          |                                            | 57 |
| 58 |          |        |          |                                            | 58 |
| 59 |          |        |          |                                            | 59 |
| 60 |          |        |          |                                            | 60 |

```
1 # SUBROUTINE: DVOVSUB
2 # AUTHOR: C. WORK, MOD 0, 12 JUNE 68
3
4 # PURPOSE: THIS SUBROUTINE PROVIDES A SINGLE-PRECISION MACHINE LANGUAGE DIVISION OPERATION WHICH RETURNS
5 # (1) THE QUOTIENT, IF THE DIVISION WAS NORMAL.
6 # (2) NEGMAX, IF THE QUOTIENT WAS IMPROPER AND NEGATIVE.
7 # (3) POSMAX, IF THE QUOTIENT WAS IMPROPER AND POSITIVE OR IF THERE WAS A ZERO DIVISOR.
8 # THE CALLING PROGRAM IS PRESUMED TO BE A JOB IN THE F BANK WHICH CONTAINS DVOVSUB. E BANK MUST BE 6.
9 # THE DIVISOR FOR THIS ROUTINE MAY BE IN EITHER FIXED OR ERASABLE STORAGE. SIGN AGREEMENT IS
10 # ASSUMED BETWEEN THE TWO HALVES OF THE DIVIDEND. (THIS IS CERTAIN IF THE A AND L REGISTERS ARE THE
11 # RESULT OF A MULTIPLICATION OPERATION.)
12 # CALL SEQUENCE: L TC DVOVSUB
13 # L +1 ADRES (DIVISOR)
14 # L +2 RETURN HERE, WITH RESULT IN A,L
15 # INPUT: DIVIDEND IN A,L (SIGN AGREEMENT ASSUMED), DIVISOR IN LOCATION DESIGNATED BY "ADRES".
16 # DIVISOR MAY BE IN THE DVOVSUB FBANK, FIXED-FIXED FBANK, EBANK 6, OR UNSWITCHED ERASABLE.
17 # OUTPUT: QUOTIENT AND REMAINDER, OR POSMAX (NEGMAX), WHICHEVER IS APPROPRIATE.
18 # DEBRIS: SCRATCHX, SCRATCHY, SCRATCHZ, A, L (NOTE: SCRATCHX, Y, Z ARE EQUATED TO MPAC +4, +5, AND +6.)
19 # ABORTS OR ALARMS: NONE
20 # EXITS: TO THE CALL POINT +2.
21 # SUBROUTINES CALLED: NONE.
```

```
22
23 DVOVSUB TS SCRATCHY # SAVE UPPER HALF OF DIVIDEND
24 TS SCRATCHX
25 INDEX Q # OBTAIN ADDRESS OF DIVISOR.
26 CA 0
27 INCR Q # STEP Q FOR PROPER RETURN SEQUENCE.
28 INDEX A
29 CA 0 # PICK UP THE DIVISOR.
30 EXTEND # RETURN POSMAX FOR A ZERO DIVISOR.
31 BZF MAXPLUS
32
33 TS SCRATCHZ # STORE DIVISOR.
34
35 CCS A # GET ABS(DIVISOR) IN THE A REGISTER.
36 AD BIT1
37 TCF ZEROPLUS
38 AD BIT1
39
40 ZEROPLUS XCH SCRATCHY # STORE ABS(DIVISOR). PICK UP TOP HALF OF
41 EXTEND # DIVIDEND.
42 BZMF GOODNEG # GET -ABS(DIVIDEND)
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

|         |        |          |                                     |
|---------|--------|----------|-------------------------------------|
|         | CS     | A        |                                     |
| GOODNEG | AD     | SCRATCHY | # ABS(DIVISOR) - ABS(DIVIDEND)      |
|         | EXTEND |          |                                     |
|         | BZMF   | MAKEMAX  | # BRANCH IF DIVISION IS NOT PROPER. |

|  |        |          |                                      |
|--|--------|----------|--------------------------------------|
|  | CA     | SCRATCHX | # RE-ESTABLISH THE DIVIDEND          |
|  | EXTEND |          |                                      |
|  | DV     | SCRATCHZ | # QUOTIENT IN THE A, REMAINDER IN L. |
|  | TC     | Q        | # RETURN TO CALLER.                  |

|         |     |          |                                       |
|---------|-----|----------|---------------------------------------|
| MAKEMAX | CCS | SCRATCHX | # DETERMINE THE SIGN OF THE QUOTIENT. |
|         | CCS | SCRATCHZ | # SCRATCHX AND SCRATCHZ ARE NON-ZERO. |
|         | TCF | MAXPLUS  |                                       |

|  |     |          |              |
|--|-----|----------|--------------|
|  | CCS | SCRATCHZ |              |
|  | CAF | NEGMAX   | # +,- OR -,+ |
|  | TC  | Q        |              |

|         |     |        |              |
|---------|-----|--------|--------------|
| MAXPLUS | CAF | POSMAX | # -,- OR +,+ |
|         | TC  | Q      |              |

# COEFFICIENTS FOR THE JET ACCELERATION CURVE FITS  
# THE CURVE FITS ARE OF THE FORM --  
#

# 1JACC = A/(MASS + C) + B

#  
# A IS SCALED AT PI/4 RAD/SEC\*\*2 B+16KG, B IS SCALED AT PI/4 RAD/SEC\*\*2, AND C IS SCALED AT B +16 KG.

#  
# THE CURVE FIT FOR L,PVT-CG IS OF THE SAME FORM, EXCEPT THAT A IS SCALED AT 8 FT B+16 KG, B IS SCALED AT 8 FT,  
# AND C IS SCALED AT B+16 KG.

|  |      |              |     |   |         |
|--|------|--------------|-----|---|---------|
|  | 2DEC | +.0410511917 | # L | A | DESCENT |
|--|------|--------------|-----|---|---------|

|          |      |              |          |   |         |
|----------|------|--------------|----------|---|---------|
| INERCONA | 2DEC | +.0059347674 | # 1JACCP | A | DESCENT |
|----------|------|--------------|----------|---|---------|

|  |      |              |          |   |         |
|--|------|--------------|----------|---|---------|
|  | 2DEC | +.0014979264 | # 1JACCQ | A | DESCENT |
|--|------|--------------|----------|---|---------|

|  |      |              |          |   |         |
|--|------|--------------|----------|---|---------|
|  | 2DEC | +.0010451889 | # 1JACCR | A | DESCENT |
|--|------|--------------|----------|---|---------|

|  |      |              |          |   |        |
|--|------|--------------|----------|---|--------|
|  | 2DEC | +.0065443852 | # 1JACCP | A | ASCENT |
|--|------|--------------|----------|---|--------|

|  |      |              |          |   |        |
|--|------|--------------|----------|---|--------|
|  | 2DEC | +.0035784354 | # 1JACCQ | A | ASCENT |
|--|------|--------------|----------|---|--------|

|  |      |              |          |   |        |
|--|------|--------------|----------|---|--------|
|  | 2DEC | +.0056946631 | # 1JACCR | A | ASCENT |
|--|------|--------------|----------|---|--------|

|  |     |          |     |   |         |
|--|-----|----------|-----|---|---------|
|  | DEC | +.155044 | # L | B | DESCENT |
|--|-----|----------|-----|---|---------|

|  |     |          |     |   |         |
|--|-----|----------|-----|---|---------|
|  | DEC | -.025233 | # L | C | DESCENT |
|--|-----|----------|-----|---|---------|

|          |     |          |          |   |         |
|----------|-----|----------|----------|---|---------|
| INERCONB | DEC | +.002989 | # 1JACCP | B | DESCENT |
| INERCONC | DEC | +.008721 | # 1JACCP | C | DESCENT |
|          | DEC | +.018791 | # 1JACCQ | B | DESCENT |
|          | DEC | -.068163 | # 1JACCQ | C | DESCENT |
|          | DEC | +.021345 | # 1JACCR | B | DESCENT |
|          | DEC | -.066027 | # 1JACCR | C | DESCENT |
|          | DEC | +.000032 | # 1JACCP | B | ASCENT  |
|          | DEC | -.006923 | # 1JACCP | C | ASCENT  |
|          | DEC | +.162862 | # 1JACCQ | B | ASCENT  |
|          | DEC | +.002588 | # 1JACCQ | C | ASCENT  |
|          | DEC | +.009312 | # 1JACCR | B | ASCENT  |
|          | DEC | -.023608 | # 1JACCR | C | ASCENT  |

|          |       |       |  |  |
|----------|-------|-------|--|--|
| GIMBLBTS | OCTAL | 01400 |  |  |
|          | OCTAL | 01000 |  |  |
|          | OCTAL | 06000 |  |  |

|         |       |         |                          |  |
|---------|-------|---------|--------------------------|--|
| DGBF    | OCTAL | 04000   |                          |  |
| 0.35356 | DEC   | 0.6     | # .3 SCALED AT 1/2       |  |
|         | DEC   | 0.35356 | # .70711 SCALED AT 2     |  |
| GFACTM  | OCT   | 337     | # 979.24/2.20462 AT B+15 |  |
| .7071   | DEC   | .70711  |                          |  |
| -.7071  | DEC   | -.70711 |                          |  |
| -EPSMAX | DEC   | -.42265 |                          |  |

## # CSM-DOCKED INERTIA COMPUTATIONS

|          |    |         |                                       |  |
|----------|----|---------|---------------------------------------|--|
| DOCKED   | CA | ONE     | # COEFTR = 1 FOR INERTIA COEFFICIENTS |  |
| SPSLOOP1 | TS | COEFCTR | # = 7 FOR CG COEFFICIENTS             |  |
|          | CA | ONE     | # MASSCTR = 1 FOR CSM                 |  |
|          | TS | MASSCTR | # = 0 FOR LEM                         |  |

|  |        |          |                |  |
|--|--------|----------|----------------|--|
|  | INDEX  | COEFCTR  |                |  |
|  | CA     | COEFF -1 | # COEFF -1 = C |  |
|  | EXTEND |          |                |  |

|  |        |         |                                   |  |
|--|--------|---------|-----------------------------------|--|
|  | MP     | LEMMASS |                                   |  |
|  | EXTEND |         |                                   |  |
|  | MP     | CSMMASS | # LET X = CSMMASS AND Y = LEMMASS |  |

|  |       |         |                    |  |
|--|-------|---------|--------------------|--|
|  | INDEX | COEFCTR |                    |  |
|  | AD    | COEFF   | # COEFF = F        |  |
|  | TS    | MPAC    | # MPAC = C X Y + F |  |
|  | TCF   | +4      |                    |  |

|          |        |         |                                       |  |
|----------|--------|---------|---------------------------------------|--|
| SPSLOOP2 | TS     | MASSCTR | # LOOP TWICE THROUGH HERE TO OBTAIN   |  |
|          | EXTEND |         | # MPAC = MPAC + (A X +D)X + (B Y +E)Y |  |
|          | DIM    | COEFCTR | # LOOP #1 LOOP #2                     |  |

|  |        |          |                     |  |
|--|--------|----------|---------------------|--|
|  | INDEX  | COEFCTR  |                     |  |
|  | CA     | COEFF +2 | # COEFF +2 = A OR B |  |
|  | EXTEND |          |                     |  |



|          |        |              |                                            |
|----------|--------|--------------|--------------------------------------------|
|          | INDEX  | MASSCTR      |                                            |
|          | MP     | LEMMASS      |                                            |
|          | INDEX  | COEFCTR      |                                            |
|          | AD     | COEFF        | +4 # COEFF +4 = E OR D                     |
|          | EXTEND |              |                                            |
|          | INDEX  | MASSCTR      |                                            |
|          | MP     | LEMMASS      |                                            |
|          | ADS    | MPAC         |                                            |
|          | CCS    | MASSCTR      |                                            |
|          | TCF    | SPSLOOP2     |                                            |
|          | CCS    | COEFCTR      | # IF COEFCTR IS POS, EXIT FROM LOOP WITH   |
|          | TCF    | +7           | # CG X DELDOT = MPAC X 4 PI RAD-CM/SEC     |
| TORQCONS | 2DEC   | 0.51443 B-14 | # CORRESPONDS TO 500 LB-FT                 |
|          | CA     | MPAC         |                                            |
|          | TS     | MPAC         | +1 # INERTIA = (MPAC +1) X 2(38) KG-CM(2)  |
|          | CA     | SEVEN        |                                            |
|          | TCF    | SPSLOOP1     |                                            |
|          | CA     | 1JACCCON     | # 1JACC=1JACCCON/MASS                      |
|          | ZL     |              |                                            |
|          | TC     | DVOVSUB      |                                            |
|          | ADRES  | MASS         |                                            |
|          | TS     | 1JACC        | # SCALED AT PI/4                           |
|          | CA     | POSMAX       | # SET INVERSE JET ACCELERATIONS TO POSMAX, |
|          | TS     | 1/ANETP      | # WHICH CORRESPONDS TO ACCEL. OF 1.4 D/SS. |
|          | TS     | 1/ANET2 +1   |                                            |
|          | TS     | 1/ANET2 +2   |                                            |
|          | TS     | 1/ANET2 +17D |                                            |
|          | TS     | 1/ANET2 +18D |                                            |
|          | EXTEND |              |                                            |
|          | DCA    | TORQCONS     |                                            |
|          | EXTEND |              |                                            |
|          | DV     | MPAC         | +1                                         |
|          | INHINT |              |                                            |
|          | TS     | 1JACCQ       | # SCALED AT PI/4                           |
|          | TS     | 1JACCR       |                                            |
|          | CA     | -.7071       |                                            |
|          | TS     | COEFFQ       | # COEFFQ AND COEFFR ARE CHOSEN TO MAKE U-  |
|          | CA     | .7071        | # AND V-AXES ORTHOGONAL FOR DOCKED CASE    |
|          | TS     | COEFFR       |                                            |
|          | CA     | MASS         | # SCALED AT 2(16) KG                       |
|          | EXTEND |              |                                            |
|          | MP     | MPAC         | # SCALED AT 4 PI RAD-CM/SEC                |
|          | EXTEND |              |                                            |
|          | MP     | ABDELV       | # SCALED AT 2(13) CM/SEC(2)                |
|          | TC     | DVOVSUB      | # GET QUOTIENT WITH OVERFLOW PROTECTION    |

|        |     |         |
|--------|-----|---------|
| -.1875 | DEC | -.18750 |
|--------|-----|---------|

|          |        |            |                                          |
|----------|--------|------------|------------------------------------------|
|          | BANK   | 20         |                                          |
|          | SETLOC | DAPS3      |                                          |
|          | BANK   |            |                                          |
|          | EBANK= | AOSQ       |                                          |
|          | COUNT* | \$\$/DAPAD |                                          |
| -1       | TS     | INGTS      | # ZERO INGTS IN ASCENT                   |
| 1/ACCONT | CA     | DB         | # INITIALIZE DBVAL1,2,3                  |
|          | EXTEND |            |                                          |
|          | MP     | BIT13      |                                          |
|          | TS     | L          | # 0.25 DB                                |
|          | AD     | A          |                                          |
|          | TS     | DBVAL3     | # 0.50 DB                                |
|          | CS     | DBVAL1     |                                          |
|          | AD     | L          |                                          |
|          | TS     | DBVAL2     | # -.75 DB                                |
| GETAOSUV | INHINT |            |                                          |
|          | CAE    | AOSR       | # COMPUTE ASOU AND AOSV BY ROTATING      |
|          | TS     | L          | # AOSQ AND AOSR.                         |
|          | CAE    | AOSQ       |                                          |
|          | TC     | IBNKCALL   |                                          |
|          | CADR   | ROT-TOUV   |                                          |
|          | DXCH   | AOSU       |                                          |
|          | RELINT |            |                                          |
|          | CA     | DAPBOOLS   |                                          |
|          | MASK   | DRIFTBIT   | # ZERO DURING ULLAGE AND POWERED FLIGHT. |
|          | CCS    | A          | # IF DRIFTING LIGHT,                     |
|          | CA     | ONE        | # SET DRIFTER TO 1                       |
|          | TS     | DRIFTER    | # SAVE TO TEST FOR DRIFTING FLIGHT LATER |
|          | AD     | ALLOWGTS   | # NON-ZERO IF DRIFT OR GTS NEAR          |
|          | CCS    | A          |                                          |
|          | CA     | FLATVAL    | # DRIFTING FLIGHT, STORE .8 IN FLAT      |
|          | TS     | FLATEMP    | # IN POWERED FLIGHT, STORE ZERO IN FLAT  |
|          | EXTEND |            |                                          |
|          | BZF    | DOPAXIS    | # IF POWERED AND NO GTS, START P AXIS,   |
|          | CCS    | DRIFTER    | # OTHERWISE SET ZONE3LIM                 |
|          | CA     | ZONE3MAX   | # 17.5 MS, SCALED AT 4 SECONDS.          |
|          | TS     | Z3TEM      |                                          |
| DOPAXIS  | CA     | 1JACC      | # 1JACC AT PI/4 = 2JACC AT PI/2 =        |
|          |        |            | # ANET AT PI/2 = ANET/ACOAST AT 2(6).    |
|          | AD     | BIT9       | # 1 + ANET/ACOAST AT 2(6)                |
|          | TS     | FUNTEM     |                                          |
|          | CA     | 1JACC      |                                          |

|          |        |             |                                            |
|----------|--------|-------------|--------------------------------------------|
|          | TC     | INVERT      |                                            |
|          | INHINT |             | # P AXIS DATA MUST BE CONSISTENT           |
|          | TS     | 1/ANETP     | # SCALED AT 2(7)/PI.                       |
|          | TS     | 1/ANETP +1  |                                            |
|          | CS     | BIT9        | # -1 AT 2(6)                               |
|          | EXTEND |             |                                            |
|          | MP     | 1/ANETP     | # -1/ANET AT 2(13)/PI                      |
|          | EXTEND |             |                                            |
|          | DV     | FUNTEM      | # -1/(ANET + ANET**2/ACOAST) AT 2(7)/PI    |
|          | TS     | PACCFUN     |                                            |
|          | TS     | PACCFUN +1  |                                            |
|          | CA     | 1/.03       | # NO AOS FOR P AXIS, ACOAST = AMIN         |
|          | TS     | 1/ACOSTP    |                                            |
|          | TS     | 1/ACOSTP +1 |                                            |
|          | RELINT |             |                                            |
|          | ZL     |             |                                            |
|          | CCS    | DRIFTER     |                                            |
|          | DXCH   | AOSU        | # ZERO AOSU,V IF IN DRIFT, JUST TO BE SURE |
| UAXIS    | CA     | ZERO        | # DO U AXIS COMPUTATIONS                   |
|          | TS     | UV          | # ZERO FOR U AXIS, ONE FOR V AXIS.         |
| BOTHAXES | TS     | SIGNAOS     | # CODING COMMON TO U,V AXES                |
|          | INDEX  | UV          |                                            |
|          | CCS    | AOSU        | # PICK UP ABS(AOSU OR AOSV)                |
|          | AD     | ONE         | # RESTORE TO PROPER VALUE                  |
|          | TCF    | +3          | # AND LEAVE SIGNAOS AT ZERO                |
|          | AD     | ONE         | # NEGATIVE, RESTORE TO PROPER VALUE        |
|          | INCR   | SIGNAOS     | # AND SET SIGNAOS TO ONE TO SHOW AOS NEG   |
|          | TS     | ABSAOS      | # SAVE ABS(AOS)                            |
|          | CS     | SIGNAOS     |                                            |
|          | TS     | -SIGNAOS    | # USED AS AN INDEX                         |
|          | CA     | DBVAL1      | # SET DB1, DB2 TO DBVAL1 (= DB)            |
|          | TS     | DBB1        |                                            |
|          | TS     | DBB2        |                                            |
|          | CA     | ABSAOS      | # TEST MAGNITUDE OF ABS(AOS)               |
|          | AD     | -.03R/S2    |                                            |
|          | EXTEND |             |                                            |
|          | BZMF   | NOTMUCH     | # ABS(AOS) LESS THAN AMIN                  |
| BIGAOS   | CCS    | FLATEMP     | # AGS(AOS) GREATER THAN AMIN               |
|          | TCF    | SKIPDB1     | # I DRIFT OR GTS, DO NOT COMPUTE DB        |
|          | CA     | DBVAL1      |                                            |
|          | INDEX  | -SIGNAOS    |                                            |

```
1
2 ADS DBB2 # DB2(1) = 2 DB
3 INDEX SIGNAOS
4 TS DBB4 # DB4(3) = 1 DB
5 CA -.1875 # -.1875 PI/2 RAD/SEC(2) SCALED AT PI/2
6 AD ABSAOS # ABSAOS IS SCALED AT PI/2
7
8 EXTEND
9 BZMF +3
10 CS DBVAL3 # -.5 DB
11
12 TCF DBONE
13 CS ABSAOS
14 DOUBLE
15 DOUBLE
16 AD BIT14
17 DOUBLE # 1-8 ABSAOS. (8 IS 16/PI SCALED AT 2/PI)
18
19 DBONE EXTEND
20 INDEX MP DB
21 TS DBB1 # DB1(2)=(1-8 ABSAOS) DB. IF ABSAOS IS
22 CA DBVAL2 # GREATER THAN .1875 THEN DB1(2) = -.5 DB
23 INDEX -SIGNAOS
24
25 TS DBB3 # DB3(4) = -.75 DB
26
27 SKIPDB1 CA ABSAOS # ABS(AOS) GREATER THAN AMIN, SO IT IS
28
29 EXTEND
30 MP BIT12
31 AD ABSAOS # (9/8) ABSAOS.
32 TC INVERT # ALL RIGHT TO DIVIDE
33 INDEX -SIGNAOS
34 TS 1/ACOSTT +1 # 1/ACOSTPOS(NET) = 1/ABS(AOS)
35
36 CA 1/.03
37 INDEX SIGNAOS
38 TS 1/ACOSTT # 1/ACOSTNEG(POS) = 1/AIN
39
40 CA ABSAOS
41 AD 1JACCU
42
43 AD 1JACCU # 2 JACC + ABS(AOS)
44 AD BIT9 # MAXIMUM VALUE IN COMPUTATIONS
45 TS A # TEST FOR OVERFLOW
46
47 TCF SKIPDB2 # NO OVERFLOW, DO NORMAL COMPUTATION
48
49 CA ABSAOS # RESCALE TO PI TO PREVENT OVERFLOW
50
51 EXTEND
52 MP BIT14
53 AD 1JACCU # 1 JACC AT PI/2 = 2JACC AT PI
54 TS ANET # ANETPOS(NEG) MAX SCALED AT PI =
55 # ANETPOS(NEG) MAX/ACOSTNEG(POS) AT 2(7)
56
57 AD BIT8 # 1 + ANETPOS/ACOSTNEG AT 2(7)
58
59 XCH ANET # SAVE IN ANET, WHILE PICKING UP ANET
60 TC INVERT
61 EXTEND
```

|          |        |             |                                          |
|----------|--------|-------------|------------------------------------------|
|          | MP     | BIT14       | # SCALE 1/ANET AT 2(7)/PI                |
|          | TS     | 1/ANET      |                                          |
|          | CA     | ACCHERE     | # SET UP RETURN FROM COMPUTATION ROUTINE |
|          | TS     | ARET        |                                          |
|          | CS     | BIT8        | # -1 AT 2(7)                             |
|          | TCF    | DOACCFUN    | # FINISH ACCFUN COMPUTATION              |
| ACCHERE  | TCF    | ACCTHERE    |                                          |
| NOTMUCH  | TS     | L           | # ABS(AOS) LESS THAN AMIN, SAVE IN L     |
|          | CA     | 1/.03       | # ACOASTPOS,NEG = AMIN                   |
|          | TS     | 1/ACOSTT    |                                          |
|          | TS     | 1/ACOSTT +1 |                                          |
|          | CCS    | FLATEMP     |                                          |
|          | TCF    | SKIPDB2     | # DO NOT COMPUTE DB IF DRIFT OR GTS      |
|          | CA     | .0125RS     | # AMIN/2                                 |
|          | AD     | L           | # L HAS ABS(AOS) - AMIN                  |
|          | EXTEND |             | # RESULT IS ABS(AOS)- AMIN/2             |
|          | BZMF   | NOAOS       | # ABS(AOS) LESS THAN AMIN/2              |
| SOMEAOS  | CA     | DBVAL3      | # AMIN/2 LT ABS(AOS) LT AMIN             |
|          | INDEX  | -SIGNAOS    |                                          |
|          | TS     | DBB3        | # DB3(4) = DB/2                          |
|          | AD     | A           |                                          |
|          | INDEX  | SIGNAOS     |                                          |
|          | TS     | DBB4        | # DB4(3) = DB                            |
|          | TCF    | SKIPDB2     |                                          |
| NOAOS    | CA     | DBVAL1      |                                          |
|          | TS     | DBB3        | # DB3,4 = DB                             |
|          | TS     | DBB4        |                                          |
| SKIPDB2  | CA     | ABSAOS      | # ANETPOS(NEG) MAX = 2 JACC + ABS(AOS)   |
|          | AD     | 1JACCU      |                                          |
|          | AD     | 1JACCU      |                                          |
|          | TS     | ANET        | # CANNOT OVERFLOW HERE                   |
| CL1/NET+ | TC     | DO1/NET+    | # COMPUTE 1/ANET, ACCFUN                 |
| ACCTHERE | INDEX  | -SIGNAOS    |                                          |
|          | TS     | Z5TEM +2    | # STORE ACCFUN IN TEMPORARY BUFFER       |
|          | CA     | 1/ANET      |                                          |
|          | INDEX  | -SIGNAOS    |                                          |
|          | TS     | 1/ATEM2 +2  | # STORE 1/ANET IN TEMPORARY BUFFER       |
|          | CA     | ABSAOS      | # SEE IF OVERFLOW IN MIN CASE            |
|          | AD     | 1JACCU      |                                          |

```
1 AD BIT9 # MAXIMUM POSSIBLE VALUE
2 TS A # OVERFLOW POSSIBLE BUT REMOTE
3
4 TCF +2
5 CA POSMAX # IF OVERFLOW, TRUNCATE TO PI/2
6 AD -.03R/S2 # RESTORE TO CORRECT VALUE
7
8 TS ANET
9 TC DO1/NET+ # COMPUTE 1/ANET, ACCFUN
10
11 INDEX -SIGNAOS # STORE MIN VALUES JUST AS MAX VALUES
12 TS Z5TEM
13 CA 1/ANET
14 INDEX -SIGNAOS
15 TS 1/ATEM2
16
17 CS ABSAOS # NOW DO NEG(POS) CASES
18 AD 1JACCU
19 AD 1JACCU # ANETNEG(POS) MAX
20 TC 1/ANET- # COMPUTE 1/ANET, ACCFUN, AND ACCSW
21 INDEX SIGNAOS # STORE NEG(POS) VALUES JUST AS POS(NEG)
22 TS Z1TEM +2
23 TS L # SAVE IN L FOR POSSIBLE FUTURE USE
24 CA 1/ANET
25 INDEX SIGNAOS
26 TS 1/ATEM1 +2
27 CS ABSAOS
28 AD 1JACCU # 1/ANETNEG(POS) MIN
29 TS ANET
30 AD -.03R/S2 # TEST FOR AMIN
31 EXTEND # IF ANET LESS THAN AMIN, STORE MAX JET
32 BZMF FIXMIN # VALUES FOR MIN JETS AND SET ACCSW
33
34 TC 1/NETMIN # OTHERWISE DO MIN JET COMPUTATIONS
35 INDEX SIGNAOS # STORE VALUES
36 TS Z1TEM
37 CA 1/ANET
38 INDEX SIGNAOS
39 TS 1/ATEM1
40
41 INDEX UV
42 CA +UMASK
43 MASK CH5MASK # TEST FOR +U (+V) JET FAILURES
44 EXTEND
45 BZF FAIL-
46 CA 1/ATEM2 # REPLACE FUNCTION VALUES DEPENDING ON THE
47 TS 1/ATEM2 +2 # FAILED JET PAIR WITH CORRESPONDING ONE-
48 CA Z5TEM # JET (OR AMIN) FUNCTION VALUES
49 TS Z5TEM +2
50 INDEX UV
51
52
53
54
55
56
57
58
59
60
```

|    |       |        |            |                                            |
|----|-------|--------|------------|--------------------------------------------|
| 1  |       |        |            |                                            |
| 2  |       | CA     | -UMASK     |                                            |
| 3  |       | MASK   | CH5MASK    | # TEST FOR -U (-V) JET FAILURES            |
| 4  |       | EXTEND |            |                                            |
| 5  |       | BZF    | DBFUN      |                                            |
| 6  |       | CA     | 1/ATEM1    | # REPLACE FUNCTION VALUES DEPENDING ON THE |
| 7  |       | TS     | 1/ATEM1 +2 | # FAILED JET PAIR WITH CORRESPONDING ONE-  |
| 8  |       | CA     | Z1TEM      | # JET (OR AMIN) FUNCTION VALUES            |
| 9  |       | TS     | Z1TEM +2   |                                            |
| 10 |       |        |            |                                            |
| 11 | DBFUN | CS     | DBB3       | # COMPUTE AXISDIST                         |
| 12 |       | AD     | DBB1       |                                            |
| 13 |       | AD     | FLATEMP    |                                            |
| 14 |       | TS     | AXDSTEM    |                                            |
| 15 |       | CS     | DBB4       |                                            |
| 16 |       | AD     | DBB2       |                                            |
| 17 |       | AD     | FLATEMP    |                                            |
| 18 |       | TS     | AXDSTEM +1 |                                            |
| 19 |       |        |            |                                            |
| 20 |       | INHINT |            |                                            |
| 21 |       | CCS    | UV         | # TEST FOR U OR V AXIS                     |
| 22 |       | TCF    | STORV      | # V AXIS STORE V VALUES                    |
| 23 |       |        |            |                                            |
| 24 |       | CA     | ACCSW      | # U AXIS STORE U VALUES                    |
| 25 |       | TS     | ACCSWU     |                                            |
| 26 |       |        |            |                                            |
| 27 |       | CA     | NINE       | # TRANSFER 10 WORDS VIA GENTRAN            |
| 28 |       | TC     | GENTRAN +1 |                                            |
| 29 |       | ADRES  | 1/ATEM1    | # TEMPORARY BUFFER                         |
| 30 |       | ADRES  | 1/ANET1    | # THE REAL PLACE                           |
| 31 |       |        |            |                                            |
| 32 |       | RELINT |            |                                            |
| 33 |       | DXCH   | DBB1       | # SAVE U DBS FOR LATER STORING             |
| 34 |       | DXCH   | UDB1       |                                            |
| 35 |       | DXCH   | DBB4       |                                            |
| 36 |       | DXCH   | UDB4       |                                            |
| 37 |       |        |            |                                            |
| 38 |       | DXCH   | AXDSTEM    |                                            |
| 39 |       | DXCH   | UAXDIST    |                                            |
| 40 |       |        |            |                                            |
| 41 |       | CA     | ONE        | # NOW DO V AXIS                            |
| 42 |       | TS     | UV         |                                            |
| 43 |       | CA     | ZERO       |                                            |
| 44 |       | TCF    | BOTHAXES   | # AND DO IT AGAIN                          |
| 45 |       |        |            |                                            |
| 46 | STORV | CA     | ACCSW      | # STORE V AXIS VALUES                      |
| 47 |       | TS     | ACCSWV     |                                            |
| 48 |       | CA     | NINE       |                                            |
| 49 |       | TC     | GENTRAN +1 |                                            |
| 50 |       |        |            |                                            |
| 51 |       |        |            |                                            |
| 52 |       |        |            |                                            |
| 53 |       |        |            |                                            |
| 54 |       |        |            |                                            |
| 55 |       |        |            |                                            |
| 56 |       |        |            |                                            |
| 57 |       |        |            |                                            |
| 58 |       |        |            |                                            |
| 59 |       |        |            |                                            |
| 60 |       |        |            |                                            |



|    |       |       |               |                                        |
|----|-------|-------|---------------|----------------------------------------|
| 1  |       |       |               |                                        |
| 2  |       | ADRES | 1/ATEM1       | # TEMPORARY BUFFER                     |
| 3  |       | ADRES | 1/ANET1 +16D  | # THE REAL PLACE                       |
| 4  |       |       |               |                                        |
| 5  |       |       |               | # NOW STORE DEADBANDS FOR ALL AXES     |
| 6  |       | DXCH  | FLATEMP       | # FLAT AND ZONE3LIM                    |
| 7  |       | DXCH  | FLAT          |                                        |
| 8  |       |       |               |                                        |
| 9  |       | CA    | DBVAL1        | # COMPUTE P AXIS DEADBANDS             |
| 10 |       | TS    | PDB1          |                                        |
| 11 |       | TS    | PDB2          |                                        |
| 12 |       | AD    | FLAT          |                                        |
| 13 |       | TS    | PDB3          |                                        |
| 14 |       | TS    | PDB4          |                                        |
| 15 |       | CA    | ZERO          |                                        |
| 16 |       | TS    | PAXDIST       |                                        |
| 17 |       | TS    | PAXDIST +1    |                                        |
| 18 |       |       |               |                                        |
| 19 |       | CCS   | FLAT          |                                        |
| 20 |       | TCF   | DRFDB         | # DRIFT OR GTS -- COMPUTE DBS          |
| 21 |       |       |               |                                        |
| 22 |       | DXCH  | UDB1          | # STORE U DEADBANDS                    |
| 23 |       | DXCH  | FIREDB        | # CANNOT USE GENTRAN BECAUSE OF RELINT |
| 24 |       | DXCH  | UDB4          |                                        |
| 25 |       | DXCH  | COASTDB       |                                        |
| 26 |       | DXCH  | UAXDIST       |                                        |
| 27 |       | DXCH  | AXISDIST      |                                        |
| 28 |       | DXCH  | DBB1          | # STORE V AXIS DEADBANDS               |
| 29 |       | DXCH  | FIREDB +16D   | # COULD USE GENTRAN IF DESIRED         |
| 30 |       | DXCH  | DBB4          |                                        |
| 31 |       | DXCH  | COASTDB +16D  |                                        |
| 32 |       | DXCH  | AXDSTEM       |                                        |
| 33 |       | DXCH  | AXISDIST +16D |                                        |
| 34 |       |       |               |                                        |
| 35 |       | TCF   | 1/ACCRET +1   | # ALL DONE                             |
| 36 | DRFDB | CA    | DBVAL1        | # DRIFT DEADBANDS                      |
| 37 |       | TS    | FIREDB        |                                        |
| 38 |       | TS    | FIREDB +1     |                                        |
| 39 |       | TS    | FIREDB +16D   |                                        |
| 40 |       | TS    | FIREDB +17D   |                                        |
| 41 |       | AD    | FLAT          |                                        |
| 42 |       | TS    | COASTDB       |                                        |
| 43 |       | TS    | COASTDB +1    |                                        |
| 44 |       | TS    | COASTDB +16D  |                                        |
| 45 |       | TS    | COASTDB +17D  |                                        |
| 46 |       | CA    | ZERO          |                                        |
| 47 |       | TS    | AXISDIST      |                                        |
| 48 |       | TS    | AXISDIST +1   |                                        |
| 49 |       | TS    | AXISDIST +16D |                                        |
| 50 |       | TS    | AXISDIST +17D |                                        |
| 51 |       |       |               |                                        |
| 52 |       |       |               |                                        |
| 53 |       |       |               |                                        |
| 54 |       |       |               |                                        |
| 55 |       |       |               |                                        |
| 56 |       |       |               |                                        |
| 57 |       |       |               |                                        |
| 58 |       |       |               |                                        |
| 59 |       |       |               |                                        |
| 60 |       |       |               |                                        |

```
1/ACCRET INHINT
 CS DAPBOOLS # SET BIT TO INDICATE DATA GOOD.
 MASK ACCSOKAY
 ADS DAPBOOLS
 RELINT
 CA ACCRETRN
 TC BANKJUMP # RETURN TO CALLER

INVERT TS HOLD # ROUTINE TO INVERT -INPUT AT PI/2
 CA BIT9 # 1 AT 2(6)
 ZL # ZERO L FOR ACCURACY AND TO PREVENT OVFL0

 EXTEND
 DV HOLD
 TC Q # RESULT AT 2(7)/PI

DOWNGTS CAF ZERO # ZERO SWITCHES WHEN USEQRJTS BIT IS UP
 TS ALLOWGTS # OR DAP IS OFF
 TS INGTS
 TCF DOCKTEST

1/ANET- ZL
 LXCH ACCSW # ZERO ACCSW
 TS ANET # SAVE ANET
 AD -.03R/S2 # TEST FOR MIN VALUE
 EXTEND
 BZMF NETNEG # ANET LESS THAN AMIN, SO FAKE IT

1/NETMIN CA ANET
 EXTEND
 INDEX -SIGNAOS
 MP 1/ACOSTT +1 # ANETNEG(POS)/ACOASTPOS(NEG) AT 2(6)

THE FOLLOWING CODING IS VALID FOR BOTH POS OR NEG
VALUES OF AOS

DO1/NET+ AD BIT9 # 1 + ANET/ACOAST AT 2(6)
 XCH ANET # SAVE AND PICK UP ANET
 EXTEND
 QXCH ARET # SAVE RETURN
 TC INVERT
 TS 1/ANET # 1/ANET AT 2(7)/PI
 CS BIT9 # -1 AT 2(6)

DOACCFUN EXTEND
 MP 1/ANET # -1/ANET AT 2(13)/PI
 EXTEND
 DV ANET # ACCFUN AT 2(7)/PI
 TC ARET # RETURN

NETNEG CS -.03R/S2 # ANET LESS THAN AMIN -- SET EQUAL TO AMIN
 TS ANET
```

```
1
2 TCF 1/NETMIN +1 # CONTINUE AS IF NOTHING HAPPENED.
3
4 FIXMIN CCS SIGNAOS
5 CA TWO # IF AOS NEG, ACCSW = +1
6 AD NEGONE # IF AOS POS, ACCSW = -1
7
8 TS ACCSW
9 AD UV # IF ACCSW = +1, TEST FOR +U (+V) JET FAIL
10 INDEX A # IF ACCSW = -1, TEST FOR -U (-V) JET FAIL
11
12 CA -UMASK +1
13 MASK CH5MASK
14 EXTEND
15
16 BZF +4
17 CS -.03R/S2 # JET FAILURE -- CANNOT USE 2-JET VALUES
18 TS ANET # ANET = AMIN
19
20 TCF STMIN- -1 # CALCULATE FUNCTIONS USING AMIN
21 CA L # L HAS ACCFUN
22 TCF STMIN- # STORE MAX VALUES FOR MIN JETS
23
24
25 # ERASABLE ASSIGNMENTS FOR 1/ACCONT
26
27
28 1/ANETP EQUALS BLOCKTOP +2
29 1/ACOSTP EQUALS BLOCKTOP +4
30 PACCFUN EQUALS BLOCKTOP +8D
31
32 PDB1 EQUALS BLOCKTOP +10D
33 PDB2 EQUALS BLOCKTOP +11D
34 PDB4 EQUALS BLOCKTOP +12D
35
36 PDB3 EQUALS BLOCKTOP +13D
37 PAXDIST EQUALS BLOCKTOP +14D
38
39
40 ACCSW EQUALS VBUF # EXECUTIVE TEMPORARIES
41
42 1/ATEM1 EQUALS ACCSW +1 # CANNOT DO CCS NEWJOB DURING 1/ACCS
43
44 1/ATEM2 EQUALS 1/ATEM1 +1 # TEMP BUFFER FOR U AND V AXES
45 1/ACOSTT EQUALS 1/ATEM1 +4
46 Z1TEM EQUALS 1/ATEM1 +6
47 Z5TEM EQUALS 1/ATEM1 +7
48
49
50 UDB1 EQUALS 1/ATEM1 +10D # UAXIS DEADBAND BUFFER
51 UDB2 EQUALS 1/ATEM1 +11D
52 UDB4 EQUALS 1/ATEM1 +12D
53 UDB3 EQUALS 1/ATEM1 +13D
54 UAXDIST EQUALS 1/ATEM1 +14D
55
56
57 DBB1 EQUALS 1/ATEM1 +16D # TEMP DEADBAND BUFFER, ALSO V AXIS
58 DBB2 EQUALS 1/ATEM1 +17D
59 DBB4 EQUALS 1/ATEM1 +18D
60 DBB3 EQUALS 1/ATEM1 +19D
61 AXDSTEM EQUALS 1/ATEM1 +20D
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

```
1 FLATEMP EQUALS 1/ATEM1 +22D
2 Z3TEM EQUALS 1/ATEM1 +23D # MUST FOLLOW FLATEMP
3
4
5 DBVAL1 EQUALS DB
6 DBVAL2 EQUALS INTB15+
7 DBVAL3 EQUALS INTB15+ +1
8
9 DRIFTER EQUALS INTB15+ +2
10
11 UV EQUALS MPAC
12 ANET EQUALS MPAC +3
13 FUNTEM EQUALS MPAC +3
14 1/ANET EQUALS MPAC +4
15 ARET EQUALS MPAC +5
16 ABSAOS EQUALS MPAC +6
17 SIGNAOS EQUALS MPAC +7
18 -SIGNAOS EQUALS MPAC +8D
19 HOLD EQUALS MPAC +9D
20 ACCRETRN EQUALS FIXLOC -1
21
22 ZONE3MAX DEC .004375 # 17.5 MS (35 MS FOR 1 JET) AT 4 SECONDS
23 FLATVAL DEC .01778 # .8 AT PI/4 RAD
24 -.03R/S2 OCT 77377 # -PI/2(7) AT PI/2
25
26 .0125RS EQUALS BIT8 # PI/2(+8) AT PI/2
27 1/.03 EQUALS POSMAX # 2(7)/PI AT 2(7)/PI
28
29 PAXISADR GENADR PAXIS
30
31 # THE FOLLOWING 4 CONSTANTS ARE JET
32 # FAILURE MASKS AND ARE INDEXED
33 -UMASK OCT 00110 # -U
34 # -V
35 +UMASK OCT 00022 # +U
36 # +V
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
1 # PROGRAM NAME: SPSRCS
2 # AUTHOR: EDGAR M. OSHIKA (AC ELECTRONICS)
3 # MODIFIED: TO RETURN TO ALL AXES VIA Q BY P. S. WEISSMAN, OCT 7, 1968
4 # MODIFIED TO IMPROVE BENDING STABILITY BY G. KALAN, FEB. 14, 1969
5 #
6 # FUNCTIONAL DESCRIPTION:
7 # THE PROGRAM CONTROLS THE FIRING OF ALL RCS JETS IN THE DOCKED CONFIGURATION ACCORDING TO THE FOLLOWING PHASE
8 # PLANE LOGIC.
9 #
10 #
11 # 1. JET SENSE TEST (SPSRCS)
12 # IF JETS ARE FIRING NEGATIVELY, SET OLDSENSE NEGATIVE AND CONTINUE
13 # IF JETS ARE FIRING POSITIVELY, SET OLDSENSE POSITIVE AND CONTINUE
14 # IF JETS ARE NOT FIRING, SET OLDSENSE TO ZERO AND GO TO OUTER RATE LIMIT TEST
15 #
16 # 2. RATE DEAD BAND TEST
17 # IF JETS ARE FIRING NEGATIVELY AND RATE IS GREATER THAN TARGET RATE, LEAVE
18 # JETS ON AND GO TO INHIBITION LOGIC. OTHERWISE, CONTINUE.
19 # IF JETS ARE FIRING POSITIVELY AND RATE IS LESS THAN TARGET RATE, LEAVE
20 # JETS ON AND GO TO INHIBITION LOGIC. OTHERWISE, CONTINUE.
21 #
22 # 3. OUTER RATE LIMIT TEST (SPSSTART)
23 # IF MAGNITUDE OF EDOT IS GREATER THAN 1.73 DEG/SEC SET JET FIRING TIME
24 # TO REDUCE RATE AND GO TO INHIBITION LOGIC. OTHERWISE, CONTINUE.
25 #
26 # 4. COAST ZONE TEST
27 # IF STATE (E,EDOT) IS BELOW LINE $E + 4 \times EDOT > -1.4$ DEG AND EDOT IS LESS THAN 1.30 DEG/SEC SET JET TIME
28 # POSITIVE AND CONTINUE. OTHERWISE, SET JET FIRING TIME TO ZERO AND CONTINUE.
29 # IF STATE IS ABOVE LINE $E + 4 \times EDOT > +1.4$ DEG AND EDOT IS GREATER THAN -1.30 DEG/SEC, SET JET TIME NEGATIVE
30 # AND CONTINUE. OTHERWISE, SET JET FIRING TIME TO ZERO AND CONTINUE.
31 #
32 # 5. INHIBITION LOGIC
33 # IF OLDSENSE IS NON-ZERO:
34 # A) RETURN IF JET TIME AS THE SAME SIGN AS OLDSENSE
35 # B) SET INHIBITION COUNTER* AND RETURN IF JET TIME IS ZERO
36 # C) SET INHIBITION COUNTER,* SET JET TIME TO ZERO AND RETURN IF SIGN
37 # OF JET TIME IS OPPOSITE TO THAT OF OLDSENSE
38 # IF OLDSENSE IS ZERO:
39 # A) RETURN IF INHIBITION COUNTER IS NOT POSITIVE
40 # B) SET JET TIME TO ZERO AND RETURN IF INHIBITION COUNTER IS POSITIVE
41 # *NOTE: INHIBITION COUNTERS CAN BE SET TO 4 OR 10 FOR THE P AND UV AXES,
42 # RESPECTIVELY, IN SPSRCS. THEY ARE DECREMENTED BY ONE AT THE BEGINNING OF
```

# EACH DAP PASS.

# THE MINIMUM PULSE WIDTH OF THIS CONTROLLER IS DETERMINED BY THE REPETITION RATE AT WHICH THIS ROUTINE IS CALLED  
# AND IS NOMINALLY 100 MS FOR ALL AXES IN DRIFTING FLIGHT. DURING POWERED FLIGHT THE MINIMUM IS 100 MS FOR THE  
# P AXIS AND 200 MS FOR THE CONTROL OF THE U AND V AXES.

# CALLING SEQUENCE:

# INHINT

# TC IBNKCALL

# CADR SPSRCE

# EXIT:

# TC Q

# ALARM/ABORT MODE: NONE

# SUBROUTINES CALLED: NONE

# INPUT:

# E, EDOT TJ, TJV, TJU TJ MUST NOT BE NEGATIVE ZERO

# OUTPUT:

# TJ, TJV, TJU

BANK 21  
SETLOC DAPS4  
BANK

COUNT\* \$\$/DAPBU

RATELIM2  
POSTHRSTEBANK= TJU  
OCT 00632 # 1.125 DEG/SEC  
CA HALFNDX AXISCTR  
TS TJUCCS OLDSENSE  
TCF POSCHECK # JETS FIRING POSITIVELY  
TCF CTRCHECK # JETS OFF. CHECK INHIBITION CTR

NEGCHECK

INDEX AXISCTR # JETS FIRING NEGATIVELY  
CS TJU  
CCS A

TC Q # RETURN

TCF +2

TCF +1

SETCTR

INDEX AXISCTR # JET FIRING REVERSAL COMMANDED. SET CTR,  
CA UTIME # SET JET TIME TO ZERO, AND RETURN

|          |        |             |                                         |
|----------|--------|-------------|-----------------------------------------|
|          | INDEX  | AXISCTR     |                                         |
|          | TS     | UJETCTR     |                                         |
| ZAPTJ    | CA     | ZERO        |                                         |
|          | INDEX  | AXISCTR     |                                         |
|          | TS     | TJU         |                                         |
| POSCHECK | TC     | Q           |                                         |
|          | INDEX  | AXISCTR     |                                         |
|          | CA     | TJU         |                                         |
| CTRCHECK | TCF    | NEGCHECK +2 |                                         |
|          | INDEX  | AXISCTR     | # CHECK JET INHIBITION COUNTER          |
|          | CCS    | UJETCTR     |                                         |
|          | TCF    | +2          |                                         |
|          | TC     | Q           | # CTR IS NOT POSITIVE. RETURN           |
|          | TCF    | ZAPTJ       | # CTR IS POSITIVE. INHIBIT FIRINGS      |
|          | TC     | Q           | # CTR IS NOT POSITIVE. RETURN           |
| UTIME    | OCT    | 00004       |                                         |
|          | OCT    | 00012       |                                         |
| OLDSENSE | OCT    | 00012       |                                         |
| NEGFIRE  | EQUALS | DAPTREG1    |                                         |
|          | CS     | ONE         | # JETS FIRING NEGATIVELY                |
|          | TS     | OLDSENSE    |                                         |
|          | CA     | EDOT        |                                         |
|          | TCF    | +4          |                                         |
| PLUSFIRE | CA     | ONE         |                                         |
|          | TS     | OLDSENSE    |                                         |
|          | CS     | EDOT        | # RATE DEAD BAND TEST                   |
|          | LXCH   | A           |                                         |
|          | CS     | DAPBOOLS    | # IF DRIFTBIT = 1, USE ZERO TARGET RATE |
|          | MASK   | DRIFTBIT    | # IF DRIFTBIT = 0, USE 0.10 RATE TARGET |
|          | CCS    | A           |                                         |
|          | CA     | RATEDB1     |                                         |
|          | AD     | L           |                                         |
|          | EXTEND |             |                                         |
|          | BZMF   | SPSSTART    |                                         |
|          | TCF    | POSTHRST +3 |                                         |
| SPSRCS   | INDEX  | AXISCTR     | # JET SENSE TEST                        |
|          | CCS    | TJU         |                                         |
|          | TCF    | PLUSFIRE    | # JETS FIRING POSITIVELY                |
|          | TCF    | +2          |                                         |
|          | TCF    | NEGFIRE     | # JETS FIRING NEGATIVELY                |
|          | TS     | OLDSENSE    | # JETS OFF                              |
| SPSSTART | CA     | EDOT        | # OUTER RATE LIMIT TEST                 |
|          | EXTEND |             |                                         |
|          | MP     | RATELIM1    |                                         |
|          | CCS    | A           |                                         |
|          | TCF    | NEGTHRST    | # OUTER RATE LIMIT EXCEEDED             |
|          | TCF    | +2          |                                         |
|          | TCF    | POSTHRST    | # OUTER RATE LIMIT EXCEEDED             |
|          | CA     | EDOT        | # COAST ZONE TEST                       |

|    |                             |        |                                                   |    |
|----|-----------------------------|--------|---------------------------------------------------|----|
| 1  |                             |        |                                                   | 1  |
| 2  |                             | AD     | E                                                 | 2  |
| 3  |                             | EXTEND |                                                   | 3  |
| 4  |                             | MP     | DKDB                                              | 4  |
| 5  |                             | EXTEND | # PAD LOADED DEADBAND. FRESHSTART: 1.4 DEG        | 5  |
| 6  |                             | BZF    | TJZERO                                            | 6  |
| 7  |                             |        |                                                   | 7  |
| 8  |                             | EXTEND |                                                   | 8  |
| 9  |                             | BZMF   | +7                                                | 9  |
| 10 |                             | CA     | EDOT                                              | 10 |
| 11 |                             | AD     | RATELIM2                                          | 11 |
| 12 |                             | EXTEND |                                                   | 12 |
| 13 |                             | BZMF   | TJZERO                                            | 13 |
| 14 | NEGTHRST                    | CS     | HALF                                              | 14 |
| 15 |                             | TCF    | POSTHRST +1                                       | 15 |
| 16 | +7                          | CS     | RATELIM2                                          | 16 |
| 17 |                             | AD     | EDOT                                              | 17 |
| 18 |                             | EXTEND |                                                   | 18 |
| 19 |                             | BZMF   | POSTHRST                                          | 19 |
| 20 | TJZERO                      | CA     | ZERO                                              | 20 |
| 21 |                             | TCF    | POSTHRST +1                                       | 21 |
| 22 |                             |        |                                                   | 22 |
| 23 | RATELIM1                    | =      | CALLCODE # = 00032, CORRESPONDING TO 1.73 DEG/SEC | 23 |
| 24 | RATEDB1                     | =      | TBUILDFX # = 00045, CORRESPONDS TO 0.101 DEG/SEC  | 24 |
| 25 |                             |        |                                                   | 25 |
| 26 | # *** END OF LMDAP .015 *** |        |                                                   | 26 |
| 27 |                             |        |                                                   | 27 |
| 28 |                             |        |                                                   | 28 |
| 29 |                             |        |                                                   | 29 |
| 30 |                             |        |                                                   | 30 |
| 31 |                             |        |                                                   | 31 |
| 32 |                             |        |                                                   | 32 |
| 33 |                             |        |                                                   | 33 |
| 34 |                             |        |                                                   | 34 |
| 35 |                             |        |                                                   | 35 |
| 36 |                             |        |                                                   | 36 |
| 37 |                             |        |                                                   | 37 |
| 38 |                             |        |                                                   | 38 |
| 39 |                             |        |                                                   | 39 |
| 40 |                             |        |                                                   | 40 |
| 41 |                             |        |                                                   | 41 |
| 42 |                             |        |                                                   | 42 |
| 43 |                             |        |                                                   | 43 |
| 44 |                             |        |                                                   | 44 |
| 45 |                             |        |                                                   | 45 |
| 46 |                             |        |                                                   | 46 |
| 47 |                             |        |                                                   | 47 |
| 48 |                             |        |                                                   | 48 |
| 49 |                             |        |                                                   | 49 |
| 50 |                             |        |                                                   | 50 |
| 51 |                             |        |                                                   | 51 |
| 52 |                             |        |                                                   | 52 |
| 53 |                             |        |                                                   | 53 |
| 54 |                             |        |                                                   | 54 |
| 55 |                             |        |                                                   | 55 |
| 56 |                             |        |                                                   | 56 |
| 57 |                             |        |                                                   | 57 |
| 58 |                             |        |                                                   | 58 |
| 59 |                             |        |                                                   | 59 |
| 60 |                             |        |                                                   | 60 |