

MEMORANDUM
RM-5437-PR
AUGUST 1967

**JOSS: ASSEMBLY LISTING
OF THE SUPERVISOR**

G. E. Bryan

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

MEMORANDUM

RM-5437-PR

AUGUST 1967

**JOSS: ASSEMBLY LISTING
OF THE SUPERVISOR**

G. E. Bryan

This research is supported by the United States Air Force under Project RAND—Contract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this Memorandum should not be interpreted as representing the official opinion or policy of the United States Air Force.

DISTRIBUTION STATEMENT

Distribution of this document is unlimited.

The **RAND** Corporation

1700 MAIN ST. • SANTA MONICA • CALIFORNIA • 90406

Published by The RAND Corporation

PREFACE

JOSS[†] is a multiuser, single-server computing system that provides for the solution of numerical problems. The system consists of a central computer containing the JOSS program and a number of typewriter consoles connected to the computer via telephone lines. The central computer turns its attention rapidly from console to console in such a way that individual users appear to have exclusive use of the system.

The JOSS supervisor is the unit that exercises overall control of the system's operation: Acting as a scheduling, resource-allocating, and synchronizing device, it ensures that all data and hardware necessary for a particular action are simultaneously available; it also meters the operation of the system to provide revenue accounting information and data describing system performance and user operations. This memorandum, an exact reproduction of the assembly listing of the JOSS supervisor, is a companion piece to RM-5216-PR, JOSS: User Scheduling and Resource Allocation⁽¹⁾ and RM-5217-PR, JOSS: Accounting and Performance Measurement,⁽²⁾ which describe in detail the operation of the supervisor.

The division of the program listing into two parts represents two assemblies: Part I is the code for the core-resident JOSS supervisor, and Part II the code for system initialization and catastrophic error recovery routines. (The latter is required only at system start-up or when serious machine errors occur; during regular JOSS operation, it is overlaid by user information.) The code is written in MACRO-6, the machine assembly language for the Digital Equipment Corporation PDP-6, on which the JOSS system is implemented. (The PDP-6 instruction set and the assembly language are described in DEC's Handbook F-65⁽³⁾ and Programming Manual: MACRO-6 Assembly Language.⁽⁴⁾)

The Index to the Program Listing enumerates the routines and data storage of the program in order of occurrence, with headings and indentations suggesting the structure of the code. Because of the limited

[†]JOSS is the trademark and service mark of The RAND Corporation for its computer program and services using that program.

ability of MACRO-6 to provide for page titling and subtitling, the heading structure of the listing itself does not always correspond to the indentations of the Index. Page numbers given in the Index refer to the pagination of this memorandum and not to those printed out by the assembler in the heading line.

This work is part of The RAND Corporation's continuing research program in computer theory and applications, particularly the implementation of man-machine systems, under U.S. Air Force Project RAND. It should be useful to those with an interest in the field of time-shared interactive terminal systems.

SUMMARY

The code listed in this memorandum provides those functions in the JOSS system that are generally described by the terms monitor, supervisor, or executive. In particular, the routines perform the following tasks:

1. *Machine interrupt handling.* These programs provide appropriate responses to hardware interrupts from the drum system, the magnetic tape unit, the master console teletype, the real-time clock, programmed operators, and machine-detected hardware errors. (Interrupts from the user consoles and the disc file are handled by the console service and the disc service routines, respectively.)
2. *Supervision, resource allocation, and scheduling.* These routines provide overall supervision of the flow of work through the JOSS system; they allocate central processor time, core space, and input/output devices to the user, and schedule the use of these resources effectively to maintain high-speed response.
3. *Monitoring.* These are various routines that maintain accounting records for allocating usage charges, that meter the overall system activity, and that record the details of individual user behavior.
4. *User log-on and log-off.* These routines provide for initial user access to the system (by recording initials, project number, and department number) and produce final charge accounting records when the user leaves the system.
5. *System initialization.* In these routines, used only at system start-up, time of day and current date are requested and recorded, and a copy of the system is saved on the magnetic drum in case system recovery is required.
6. *Machine error recovery.* These routines provide for recovery from catastrophic machine failure by attempting to reload the system from the copy saved at initialization time. To aid in failure diagnosis, a message printed at the master console indicates the immediate symptom of the failure and records the contents of certain important registers.

CONTENTS

PREFACE	iii
SUMMARY	v
INDEX TO THE PROGRAM LISTING	ix
PART I. JOSS SUPERVISOR	1
PART II. PREAMBLE AND RECOVERY	153
REFERENCES	175
JOSS BIBLIOGRAPHY	177

INDEX TO THE PROGRAM LISTING

PART I. JOSS SUPERVISOR

Parameters

Supervisor Assembly Parameters	4
System Parameters	4
Assembly Parameters for the PI Channels	4
Register Assignments	5
Assignments for Pseudo Data Switches	6
Macro and Operation Definitions	7
Interns and Externs	8

Tables, Lists, Queues, and Counters

Supervisor Storage	9
System Storage	9
State Queue Entry Words	9
Station Status Cells	10
Core Map	10
Assignments for I/O Buffers	11
Monitor Switches	12
User State Queue Headers	13
State Action Table for Interpreter Entry	14
Log-on Scanner Storage	15
Counters for Accounting, Statistics, and Debugging	16
Header Line for Console TTY Output	18
Storage for Summary Data	19
Per User Counters	21
Field Size, Scale Factor, and Pointer for Counters	22

I/O Interrupt Routines

Data Control Interrupt Routine	24
Drum Channel Interrupt Routines	25
Write End Routine	25

Interrupt End Sequence	26
Read End Interrupt	27
Initiate Next Write	28
Drum Error Recovery	29
Checksum a User's Block	31
Initiate Transfer to Drum	32
Data for Drum Transfers	33
Initiate Transfer from Drum to Core	34
Tape Interrupt Routines	35
Mag Tape Output Start Routine	37
Tape Restart Routine	37
Console TTY Interrupt Routines	38
Start Routine for Console TTY Output	38
Input Interrupt from Console TTY	39
User Operation (UUO) Interpreter	40
Handle Machine Failure Halts--Try To Recover	41
Halt Routine	41
Attempt Recovery from the Drum	41
Processor Interrupt Routines	42
Channel 7 Interrupts	42
Detect Indirect Address Loops	44
Constants	45
Processor Error Recovery	46
Discard a Bad User Block	46
Output a GRONK Message	47
Convert Routines	49

Interpreter Entry Processing

Monitor-Interpreter Interface	50
Decode Interpreter Signals	50
Interpreter Signal Types	50
Return to Interpreter	50
Interpreter-Monitor Entry Routines	51
Computation Break	51
Transmit Line to User	51

Switch Console to User	51
Wait for Signals and Time	52
Delay User for (E) Seconds	52
Get a Buffer	53
Return a Buffer	53
Final "Off" for User	54
Send Page Heading	55
Make Up Page Heading Message	56
Disc Entry Routines	57
Disc Action Routines	57
Disc Restart Routine	57
Produce Disc Accounting Record	58
Conversion Routine--Binary to Base Ten ASCII	59
Return Blocks of Core	60
Provide More Core for User's Block	61
Imperative Request for More Core	61
Find a Free Block of Core	62
Find a Free Block of the Required Size	63
Find a Free Block Below User	63
Find a Free Block Above User	64
Move User to Free Area	65
Move Down Multiple Users	65
Move User(s) Up	66
Reset User Locations	67
Block Transfer Core	68
Set Relocation Register	68
Compact Core	69
Do One Compact	69
Find a Core Move	70
 <i>Swap Selection</i>	
Select a User for Drum Swap	71
Find a User To Bring In	71
Find a Low Priority User of the Proper Size	72
Select Multiple Users for Transfer to Drum	73

Compact Core for Swap	75
Search User States	76

Main Processing Loop Routines

Miscellaneous MPL Functions	77
Force Users Off	77
Process Users Waiting for Buffers	78
Process the Pause Queue	78
Process Disc Interrupt Signal	79
Search for GRONKed Users	79
Inter-console Synchronizing Signals	80
Get Signal Table Index	80
Reset a Signal Bit	80
Test a Signal Bit	80
Set a Signal Bit	81
Table of Inter-console Signals	81
Disc Accounting Routines	82
Midnight Disc Skulker	82
Start or Restart Skulk	82
Disc Action Complete	83
Shutdown Procedure	84

Console Signal Processing

Monitor-Distributor Interface	85
Process Distributor Signals	85
Tables for Signal Interpretation	86
Send Messages to All in the Queue	87
Off-signal Routines	88
On-signal Routines	89
Send a Queue Message	90
Convert to Two Decimal Characters	91
Compute Binary Seconds from Clock Cells	91
In-signal Routines	92
Carrier-return-signal Routines	93

Convert to Linear Range	93
Transmission-over-signal Routines	94
Switch Console to Green State	95
<i>Accounting and Statistics Routines</i>	
Accumulate Counts on the Minute	96
Display User's Initials and Station Number	97
Display Initials for On and Off Signals	98
Convert Station Number to Octal ASCII	98
Display Statistical Distributions	99
Count Time and Gather Statistics	100
New Minute	100
New Hour	101
Tape Output Routine	102
Output Cumulative Statistics Line	103
Output Hourly Heading Line	104
Output Detail Statistics on the Minute	105
Routine To Update the Date	106
Convert Date and Time to ASCII	106
Format a Counter for Console Output	107
<i>Main Processing Loop</i>	
Monitor Main Processing Loop	108
Master Monitor Entry Point	108
Process the Queue for Transfer to Drum	108
Select an In-core User for Interpretation	109
Monitor Idle Loop	109
Process an "On" User	110
Initialize for Interpretation	111
<i>Log-on and Log-off Processors</i>	
JOSS Log-on Processor	112
Routine To Approve Initials	115
Scan Table for Approval of Initials	116

Routine To Approve Job Number	117
Tables of RPNs	118
Scan Table for Job Number Approval	119
Routine To Approve Department Name or Number	120
Scan Table for Department Approval	121
Department Name-Number Table	122
String Scanner	123
Character To Type Conversion Table	124
JOSS Log-off Processor	125

Service Routines

Change User State	127
Output Buffer to User	128
Get a Buffer from Available List	129
Return a Buffer to Available List	130
Send a Message to the Console TTY	131
Stuff Message in Buffer	132
Log-on and Shutdown Messages	133
Literal Table	135
Symbol Table	136

PART II. PREAMBLE AND RECOVERY

Preamble

Assembly Parameters	156
Interns and Externs	157
Dump Core on Tape	158
Routine To Input Date and Time	159
Ask about DDT and Allocate User Blocks	161
Write and Read Console TTY	162

Recovery

Recovery and Error Dump Routine	163
Clear Machine Status	164

Output Error Message and Panel Dump	165
Output Formatting Routines	166
Restore Date, Time, Etc.	167
Low Memory Contents	168
Low Memory Cells	169
Read and Write JOSS on Drum	170
JOSS Initialization	171
Symbol Table	172

PART I
JOSS SUPERVISOR

00040 ; JOSS SUPERVISORY UNIT
00050 ;
00060 ; G. E. BRYAN
00070 ;
00080 ;
00090 ; THIS CODE PROVIDES FOR THE JOSS SYSTEM THOSE FUNCTIONS
00100 ; THAT ARE GENERALLY DESCRIBED BY THE TERMS EXECUTIVE, MONITOR,
00110 ; OR SYSTEM SUPERVISOR. IN PARTICULAR, THESE ROUTINES PROVIDE
00120 ; FOR HANDLING OF I/O INTERRUPTS (FROM DRUM, TAPE, CONSOLE
00130 ; TTY, AND CENTRAL PROCESSOR). RECOVERY FROM MACHINE
00140 ; ERROR, SCHEDULING OF PROCESSOR TIME AND CORE SPACE, MONITOR-
00150 ; ING OF USER AND SYSTEM PERFORMANCE, AND THE LOG-ON AND LOG-
00160 ; OFF PROCESSES.
00170 ;
00180 ; ASSEMBLED 8/1/67 FROM TAPE 56 (A SEQUENCED COPY OF 31)
00190 ; BINARIES ARE ON TAPE 24, CALLED SU.
00200 ;
00210 ;
00220 PATCH: BLOCK 20
00230

```

00250 ; SYSTEM PARAMETERS
00260 N.PP1=20; LENGTH OF THE PUSH-POP LINKAGE LIST
00270 N.S=D40; ASSEMBLED NUMBER OF STATIONS
00280 S.M=2; NUMBER OF 32 BIT GROUPS FOR SIGNAL TABLES
00290 ; (FP(N.S/32)=0;N.S/32;IP(N.S/32)+1)
00300 N.SG=N.S; LENGTH OF THE SIGNAL TABLE
00310 BUFSIZ=1D18; BUFFER SIZE # OF CORE BLOCKS AVAILABLE TO USERS
00320 N.CCR=1D16; BBLCK=1D16; BEGINNING USER BLOCK
00330 ; NUMBER OF BUFFERS ASSEMBLED THE NUMBER OF TICKS PER SECOND.
00340 DELTA=4 K2=N.S+DELTA DISPLAY INITIALS IF FP(MIN/K3)=0
00350 00360 K1=1D60;
00370 K3=1D15;
00380 SA=12.
00390
00400
00410 ; ASSEMBLY PARAMETERS FOR THE PI CHANNELS
00420
00430 CDC=1; 1 - DATA CONTROL
00440 CDRM=2; 2 - I/O PROCESSOR AND DRUM CHANNEL
00450 CI630=3; 3 - 630 INPUT
00460 CO630=4; 4 - 630 OUTPUT
00470 CDAT=6; INTERRUPT CHANNEL FOR TAPE
00480 CCTY=6; 6 - CONSOLE TELETYPE
00490 CAPR=7; 7 - PROCESSOR INTERRUPTS
00500
00510 CHDC=100; THESE ARE CORRESPONDING DEFINITIONS FOR PI CONOS
00520 CHDRM=40;
00530 CH630=30
00540 CHDAT=0
00550 CHCTY=2
00560 CHAPR=1
00570 MTC=220;
00580 MT1=224
00590 MT2=230
00600 DR=400
00610 DP=010
00620 DMT=20;

```

DEVICE CODES FOR MAG TAPE REGISTERS

DATA CONTROL DEVICE NUMBER FOR MAG TAPE

	REGISTER ASSIGNMENTS
00640	
00650	A=0
00660	B=1
00670	C=2
00680	D=3
00690	E=4
00700	F=5
00710	G=6
00720	H=7
00730	I=10
00740	J=11
00750	K=12;
00760	L=13;
00770	M=14;
00780	N=15;
00790	S=16;
00800	USED BY DISTRIBUTOR USED FOR STATION INDEX PUSH-POP REGISTER
00810	PP=17;

00830 , PSEUDO DATA SWITCHES (IN SWITCH)
00840 , RIGHT HALF
00850 ,
00860 SDS=200000; SHUTDOWN SWITCH
00870 BEEPS=400000; BEEP SWITCH
00880 OFFS=100000; BLAST OFF SWITCH
00890 DCLOB=40000; DISC NOT AVAILABLE
00900 HALTS=20000; HALT ON ERROR SWITCH
00910 OLSW=100; OUTPUT STAT LINE (NUMBER IN LEFT HALF)
00920 PRMES=40; ADMINISTRATIVE MESSAGE HAS PRIORITY
00930 DSTAT=20; DISPLAY STATISTICAL DISTRIBUTIONS
00940 DOAF=10; DISPLAY ONS AND OFFS
00950 TPO=4; DISABLE MAG TAPE OUTPUT
00960 DUI=2; DISPLAY USERS INITIALS
00970 DAM=1; DISPLAY ADMINISTRATIVE MESSAGE
00980
00990 ; LEFT HALF
01000 ;
01010 ;
01020 ;
RIGHT 6 BITS ARE STAT LINE NUMBER FOR DISPLAY

```
01040      DEFINE FSW (A)
01050      <MOVE A,SWITCH
01060      >
01070      DEFINE XMT (M,A) ; TRANSMIT M CHARACTERS FROM A
01080      <OPDEF X [M'B12]
01090      X A>      DEFINE ZBLOK (N)
01100      <REPEAT N,
01110      <Z
01120      XLIST>
01130      LIST>
01140
01150
01160      DEFINE QCT (Q,C) ; MACRO TO COUNT QUEUES
01170      <
01180      HRRZ B,Q
01190      TSX QCTR
01200      MOVE M A,C>
01210
01220      OPDEF      INS [10B8]
01230      OPDEF      CHS [20B8]
01240      OPDEF      SEND [30B8]
01250      OPDEF      HALT [40B8]
01260      OPDEF      TALLY [AOS 0]
01270      OPDEF      NOP [JFCCL]
01280      OPDEF      DONE [POBJ PP,0]
01290      OPDEF      TSX [PUSHJ PP,0]
01300      OPDEF      CALL [PUSHJ PP,0]
```

```
01320      EXTERN C20,C27,C28,T80,T80.99
01330      EXTERN FILE,KEY,PROG
01340      EXTERN DBUF
01350      EXTERN DISC.D,ACTION,RESULT
01360      EXTERN C30,C31,DISC,DISC.C,DISC.S
01370      EXTERN SHUT
01380      EXTERN INTENT,INTBEG,SEQ,RISIG,INITIALS,JOBNO
01390      EXTERN PAGNO,ONTIME,COMTIM
01400      EXTERN USIZE
01410      EXTERN SPARE1,SPARE2,CONSOL
01420      EXTERN SPARE3,SPARE4
01430      EXTERN DDT,D.TIME,DUMP,DATIME,RJD,WJD
01440      EXTERN INTENT; LOC. IN USER AREA TO START THE PROG.
01450      EXTERN SWITCH,FAKE
01460      EXTERN C32
01470
01480      INTERN DATE,HR,MIN,TIME
01490      INTERN CTD,CTDRL,CTDB
01500      INTERN CT24,CT17,CT23,CT37
01510      INTERN CT14,CT19,CT18
01520      INTERN CT15,CT22
01530      INTERN CT27A,CT27B,CT46,CT47
01540      INTERN CT22A,CT18A,CT18B,TERASE,S.S,N.CB
01550      INTERN YEAR,MONTH,DAY,SEC,CT22A
01560      INTERN S.OK
01570      INTERN N.SON,APRR,PROP
01580      INTERN CT11,CT13,CT16,CT28,CT31
01590      INTERN DRMR,S.QUE,CMESS,ADATE,OCTW
01600      INTERN T8,KILL
01610      INTERN S.BUF,COMEBACK,N.S,SIGTBL,SG.L,SG.M,SG.LIM
01620      INTERN USERS,MONENT,T7.9
01630      INTERN CORE,SECONDS,CTYR
01640      INTERN HMES,PPW,TYPE6,SOUT,SIGPR
01650      INTERN SSIG,RSIG,TSIG
01660      INTERN T9,T10,T11,T11.1,T11.2,T11.3,T11.4,T11.5
01670      INTERN T11.6,T11.7,T11.8,T11.9
01680
```

01700	;	SYSTEM STORAGE	
01710	PPSAV: Z; CUI: Z; TYPE6: XWD DATE: ASCII	\$1/17/67	MONITOR PUSH-POP REGISTER SAVE CURRENT USER # IN RIGHT HALF CONTROL WORD FOR INITIALIZATION RECORD
01720			
01730			
01740			
01750			
000024	331564 02100		
000025	311467 232556	ATIM: ASCII \$23:57\$	
000026	000000 000000	YEAR: DEC 0	
000027	000000 000001	MONTH: DEC 1	
000030	000000 000021	DAY: DEC 17	
000031	000000 000027	HR: DEC 23	SECONDS IN THIS MINUTE
000032	000000 000071	MIN: DEC 57	TICKS IN THIS SECOND
000033	000000 000000	SEC: Z; TICK: Z;	REG C - PROCESSOR INTERRUPT ROUTINE
000034	000000 000000	CL9: Z;	REG D
000035	000000 000000	CL10: Z;	INDIRECT ADDRESS CHAIN COUNT
000036	000000 000000	CL11: Z;	LOCATION COUNTER
000037	000000 000000	CL12: Z;	COUNT OF INTERRUPTS AT SAME LOCATION
000040	000000 000000	CL13: Z;	CURRENT TIME IN TICKS. (16 2/3MS=1/60 SEC)
000041	000000 000000	TIME: Z;	TIME USED BY CURRENT USER THIS SHOT.
000042	000000 000000	T-CU: Z;	
000043	000000 000000	SECONDS:Z;	TICKS--CONTINUOUS COUNT FOR TIMER COMPUTATION
000044	000000 000000	SIGTBL: REPEAT N.SG.;	DISTRIBUTOR-MONITOR SIGNAL LIST
		<2	
		XLIST>	
000045	000000 000000	Z	
		XLIST	CURRENT NUMBER OF ENTRIES IN SIGTBL
000115	000000 000000	SG.L: Z; SG.M: DEC 5;	MAX # ENTRIES IN SIGTBL
000116	000000 000050	SG.LIM: DEC 5;	SIGTBL ENTRIES BEFORE COMEBACK (EXCEPT CR)
000117	000000 000005		
000120	777760 000120*	PPW: XWD PP1: REPEAT N.PP1,;	PUSH-POP LINKAGE LIST
		<2	
		XLIST>	
000121	000000 000000	Z	
		XLIST	STATE QUEUE ENTRY WORDS
		02060 ;	
		02070 ;	
		02080 S.Q: REPEAT N.S-1,	
		<XWD 0..+1 XLIST>	
		02090 02100	
000141	000000 000142*	XWD 0..+1 XLIST	
000210	000000 000000	02120 Z;	LAST QUEUE ENTRY WORD

02140	*	STATION STATUS CELLS				
02150	*	XLIST				
02160	S.S.	REPEAT N.S.				
02170	<BYTE (6) OF S.0.0.1,0,0					
02180	XLIST>					
02190	BYTE (6) OF S.0.0.1,0,0					
000211	230000 010000					
000261	360616 000211*	02210	S. STA:	POINT 6,S. S(S),5;	CURRENT STATE	(5 BITS)
000262	300616 000211*	02220	S. TM:	POINT 6,S. S(S),1D11;	TIME IN SHOT	(4 BITS)
000263	220616 000211*	02230	S. COR:	POINT 6,S. S(S),1D17;	CORE LOCATION	(5 BITS)
000264	140616 000211*	02240	S. BLOCK:	POINT 6,S. S(S),1D23;	NUMBER OF BLOCKS	(2 BITS)
000265	130116 000211*	02250	S. OFR:	POINT 1,S. S(S),1D24;	OFF SIGNAL RECEIVED	
000266	120116 000211*	02260	S. INR:	POINT 1,S. S(S),1D25;	IN SIGNAL RECEIVED	
000267	110116 000211*	02270	S. DU:	POINT 1,S. S(S),1D26;	USING DISC	
000270	100116 000211*	02280	S. GK:	POINT 1,S. S(S),1D27;	SET WHEN USER IS GRONKED	
000271	070116 000211*	02290	S. SIG:	POINT 1,S. S(S),1D28;	SET WHEN WAITING FOR SIGNALS	
02300	CORE MAP					
02310	*					
02320	CORE:	ZBLOK N.COR; THE CORE MAP				
02330	REPEAT N.CCR,					
000272	000000 000000	2	N. CB:	DEC 4;	MAX # OF CORE BLOCKS PER USER	
000312	000000 000004	02340	N. C:	DEC N.COR;	# OF ACTIVE CORE BLOCKS (HIGHEST # NOT TO USE)	
000313	000000 000020	02350	S. IU:	POINT 1,CORE(B),0;	USE BIT FOR BLOCK	
000314	440101 000272*	02360	S. ID:	POINT 1,CORE(B),1;	IN USE BY DRUM	
000315	420101 000272*	02370	S. UD:	POINT 2,CORE(B),1;	BOTH ACTIVITY BITS	
000316	420201 000272*	02380	S. UR:	POINT 6,CORE(B),1D8;	USER NUMBER FOR THE BLOCK	
000317	330601 000272*	02390				
02400	02410	*	SYSTEM PARAMETERS			
000320	000000 000000	02420	N.DRM:	Z;	NUMBER OF USERS ON DRUM	
000321	000000 000020	02430	DBASE:	OCT 20;	BASE TRACK FOR USER BLOCKS	
000322	000000 000001	02440	N.UC:	DEC 1;	START COMPUTING IF ONLY THIS MANY BUFFERS	
000323	000000 000002	02450	N.CK:	DEC 2;	QUIT COMPUTING IF THIS MANY BUFFERS FILLED	
000324	000000 000050	02460	N.SON:	DEC 40;	MAX # OF USERS BEFORE QUEUE MESSAGE	
000325	000000 000047	02470	S.OK:	OCT 47;	IGNORE SIGNALS FROM HIGHER STATIONS	
000326	000000 000014	02480	T.MAX:	DEC 12;	MAXIMUM SHOT TIME • 200 MS.	

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
ASSIGNMENTS FOR I/O BUFFERS

PAGE 9

000327	000000 000000	02510	DT.BUF:Z;	HEADER FOR DISC AND TAPE I/O
000330	000000 000000	02520	CO.BUF: Z;	HEADER FOR CONSOLE TTY OUTPUT
		02530	S.BUF: ZBLOK N.S;	BUFFER HEADERS: RIGHT = FIRST; LEFT = LAST
	REPEAT N.S.			
000331	000000 000000	Z		
000401	000000 000054	02540	N.BUF: DEC	NUMBER OF AVAILABLE BUFFERS.
000402	002064 000403*	02550	L.BUF: XWD	HEAD OF AVAILABLE BUFFER LIST
		02560		
	02570	*	LASTB,BUF;	
	02580		THE I/O BUFFERS	
	02590	BUF:	REPEAT K2-1,	
		<XWD	0..+1+BUFSIZ	
		02600	BLOCK	
		02610	BUFSIZ	
		02620	XLIST>	
		XWD	0..+1+BUFSIZ	
			BLOCK	
			BUFSIZ	
		XLIST		
000403	000000 000426*	02640	LASTB: XWD	0,0
002064	000000 000000	02650	BLOCK	BUFSIZ;
				THE LAST BUFFER

MONITOR SWITCHES

002107	000000	000000	02670	DMBY: Z;	SET WHEN DRUM IS BUSY
002110	000000	000000	02680	DKBY:Z;	SET WHEN DISC IS BUSY
002111	000000	000000	02690	COMEBACK:Z;	SET TO REQUEST INTERPRETER RETURN TO MONITOR
002112	000000	000000	02700	MISC: Z;	SWITCH USED BY XMTR, DCOMP, TRST, ADIS, RUL
002113	000000	000000	02710	CSS: Z;	SWITCH FOR CHS TO TOP OF LIST
002114	000000	000000	02720	CKF: DEC	SET TO DO CHECKSUMS OF THE DRUM

02740	02750	DEFINE STAC (A)	
02760	<IRP (A) <A> *S=A-S-QUE>		
02770	'	DEFINITIONS FOR THE STATE INDEXES.	
02780	'	END. S=37; END SIGNAL FOR PARTIAL ORDERING QUEUES	
02790			
02800	02810	S. QUE=.	THE ORDER OF THE TABLE REPRESENTS SWAP PRIORITIES
	02820	TOP: Z;	WAITING FOR TERMINAL "OFF" PROCESSING
	02830	ON: Z;	WAITING FOR "ON" PROCESSOR
002115	000000	000000	CARRIAGE RETURN
002116	000000	000000	REQUEST IN (INTERRUPT BUTTON)
002117	000000	000000	REQUEST IN--NEEDS A BUFFER
002120	000000	000000	UNCHOKE
002121	000000	000000	QUEUE FOR MORE CORE
002122	000000	000000	COMPUTE
002123	000000	000000	
002124	000000	000000	
002125	000000	000000	CURRENT USER
	02920	CU: Z;	
	02930		WAITING FOR DISC CONTINUE
	02940	DCT: Z;	DISC ACTION IN PROGRESS
	02950	DIP: Z;	CHORED ON OUTPUT
	02960	CK: Z;	QUEUE FOR ACCESS TO THE DISC
	02970	DO: Z;	GREEN (CONSOLE INPUTTING)
	02980	GR: Z;	DRAINING BUFFERS FOR SWITCH TO USER
	02990	DSU: Z;	AWAITING A BUFFER FOR GREEN
002133	000000	000000	PAUSE QUEUE
002134	000000	000000	QUEUE FOR TRANSFER TO DRUM
002135	000000	000000	QUEUE FOR JOSS INITIAL SERVICE
002136	000000	000000	
002137	000000	000000	
002140	000210.000141*	03040 OF: XWD	S. Q+N. S-1, S. Q: STATION OFF
	03050		
	03060		
	03070	IRP	STAC <TOF, ON, RC, RI, RIB, UC, QC, COM, CU>
	03080	IRP	STAC <DCT, DIP, CK, DQ, GR, DSU, ABG, QR, QDM, QM, OF>
	03090		ALSO STATE RELATED TABLES AT T1,T10. SI1, S10, BOF10.
	03100		S10,S11,S12

		ACTION BIT DEFINITIONS	
03120			
03130	CORBIT=20;	CORBIT	CORE BIT
03140	DKBIT=10;	XWD	DISC BIT
03150	RCBIT=1;	XWD	CARRIER RETURN BIT
03160	INBIT=2;	XWD	INTERRUPT BIT
03170	BUFBIT=4;	XWD	BUFFER REQUEST BIT
03180	ONBIT=40;	XWD	ON STATE BIT
03190			
03200			
03210			
03220			
03230	T10. :	XWD	0, EJ; TOF
		XWD	ONBIT, BJ; ON
002141	000000 010151*	03240	RCBIT, INTENT; RC
002142	000040 007462*	03250	INBIT, INTENT; RI
002143	000001 000000	03260	BUFBIT+INBIT, INTENT; RIB
002144	000002 002143*	03270	BUFFBIT, INTENT; UC
002145	000006 002144*	03280	CORBIT, INTENT; QC
002146	000004 002145*	03290	0, INTENT; COM
002147	000020 002146*	03300	
002150	000000 002147*	03310	

LOG-ON SCANNER STORAGE

002151	440700	000000	03330	SCP1:	POINT	7,0;
002152	440401	010131	03340	SCP2:	POINT	4,CART (B) ;
002153	440700	000000	03350	SCP3:	POINT	7,0;
002154	360600	002152	03360	SCP4:	POINT	6,SCP2,5;
002155	255001	000000	03370	SC9:	NOP	0,0 (B) ;
			03380			

INPUT BYTE POINTER
CODE TABLE BYTE POINTER
OUTPUT BYTE POINTER
POINTER TO POINTER POSITION PART
POINTER TO EXECUTE TABLE

002156	000000	000000	03400	CT10: Z;
002157	000000	000000	03410	CT10A: Z;
002160	000000	000000	03420	CT32: Z;
002161	000000	000000	03430	CT33: Z;
002162	000000	000000	03440	CT2=: Z;
002163	000000	000000	03450	CT34: Z;
002164	000000	000000	03460	CT35: Z;
002165	000000	000000	03470	CT28: Z;
002166	000000	000000	03480	CT13: Z;
002167	000000	000000	03490	CT14: Z;
002170	000000	000000	03500	CT15: Z;
002171	000000	000000	03510	CT27: Z;
002172	000000	000000	03520	CT16: Z;
002173	000000	000000	03530	CT37: Z;
002174	000000	000000	03540	CT31: Z;
002175	000000	000000	03550	CT2A=: Z;
002176	000000	000000	03560	CT25: Z;
002177	000000	000000	03570	CT26: Z;
002200	000000	000000	03580	CT27A: Z;
002201	000000	000000	03590	CT27B: Z;
002202	000000	000000	03600	CT12: Z;
002203	000000	000000	03610	CT11: Z;
002204	000000	000000	03620	CT39: Z;
002205	000000	000000	03630	CT19A: Z;
002206	000000	000000	03640	CT18: Z;
002207	000000	000000	03650	CT18A: Z;
002210	000000	000000	03660	CT18B: Z;
002211	000000	000000	03670	CT19: Z;
002212	000000	000000	03680	CT17: Z;
002213	000000	000000	03690	CT23: Z;
002214	000000	000000	03700	CT43: Z;
002215	000000	000000	03710	CT44: Z;
002216	000000	000000	03720	CT3A=: Z;
002217	000000	000000	03730	CT46: Z;
002220	000000	000000	03740	CT47: Z;
			03750	CT48A: Z;
			03760	CT48B: Z;
			03770	CT48C: Z;

MINUTE
HOUR
U - USERS
GQ - GREEN STATION QUEUE

CQ - COMPUTE QUEUE
BQ - BUFFER, OR CHOKE QUEUE
Q - THE QUEUE
COM - CHARGED COMPUTE TIME
ST - STATEMENTS INTERPRETED
A - UNUSED
I - IN REQUESTS
TL - LINES TRANSMITTED TO USERS
B - BACKSPACES RECEIVED FROM USERS
CR - LINES RECEIVED FROM USERS

1 - COUNTER ONE
2 - COUNTER TWO
3 - COUNTER THREE
4 - COUNTER FOUR
U - UNOVERLAPPED I/O TIME (DRUM)
RP - IDLE LOOP COUNT
IC - USERS IN CORE
SW - DRUM SWAPS
S - SAVES
D - DELETES
DI - PRINT DICTIONARY
L - LOADS FROM DISC
SR - STATUS RESPONSES
SE - STATUS ERRORS

1/100 TH % OF AVAIL TIME SPENT ON USERS

TOTAL COMPUTE TIME FOR USERS SINCE BEGIN

INPUT INTERRUPTS

OUTPUT INTERRUPTS

CELLS FOR DISPLAY OF SIZE DISTR.

	CT7=	COUNT CORE COMPACTS
002221	000000 000000	T - TAPE REWRITE TRIES
002222	000000 000000	E - TAPE ERASES
002223	000000 000000	- - UNERASABLE TAPE
002224	000000 000000	D - DRUM ERRORS
002225	000000 000000	K - DISC ERRORS
002226	000000 000000	C - CONSOLE DETECTED PARITY ERRORS
002227	000000 000000	S - SCANNER DETECTED PARITY ERRORS
002230	000000 000000	
03790	03800	
	03810	E - TAPE ERASES
	03820	- - UNERASABLE TAPE
	03830	D - DRUM ERRORS
	03840	K - DISC ERRORS
	03850	C - CONSOLE DETECTED PARITY ERRORS
	03860	S - SCANNER DETECTED PARITY ERRORS
	03870	
	03880	
	03890	CT49=N. DRM
	03900	TERROR=CT41
	03910	TERASE=CT42
	03920	USERS=CT32
	03930	

-17-

			HEADER LINE FOR CONSOLE TTY OUTPUT	
002231	522324	052644	03950 ;	03960
002232	266172	126606		HEAD: ASCII /TM UR-GQ-C-B COM-STA-A I--TL-C/;
002233	266044	041636		30 CHAR
002234	465332	352202		
002235	266024	044532		
002236	266511	426606		
002237	511330	344532	03980	ASCII /R-CI-CO T-K-T-#-R-#-D TX-RP-U-IC SW-S-L-D/; 42 CHAR
002240	416364	052132		
002241	455332	426506		
002242	266445	521532		
002243	421012	422532		
002244	512405	552532		
002245	446064	020246		
002246	535332	326630		
002247	266100	000000		

04000 ; STORAGE FOR SUMMARY DATA
04010 ; BLOCKS FOR DISTRIBUTIONAL DATA
04020 ; COUNTER RANGES READING THE DISPLAY FROM THE LEFT ARE:
04030 ; >500,500-200,200-100,100-50,50-20,20-10,10-5,5-2,2-1,<1
04040 ;
04050 ;
04060 ; UNITS FOR THE DISPLAYS ARE:
04070 ; SESSION TIME - MINUTES
04080 ; COMPUTE TIME - 10 SECONDS
04090 ; PROGRAM SIZE - CELLS USED
04100 ; GR-GR TIME - 1/10 MINUTES
04110 ; COMP/INTERAC - TICKS
04120 ;
04130 ;
04140 ; SESTIM: ZBLOK 12; CUMULATIVE SESSION TIME DISTRIBUTION
REPEAT 12.
002250 000000 000000 Z 04150 CCTIM: ZBLOK 12; CUMULATIVE COMPUTE TIME
REPEAT 12.
002262 000000 000000 Z 04160 DSIZ: ZBLOK 12; * SIZE
REPEAT 12.
002274 000000 000000 Z 04170 DTIM: ZBLOK 12; * GREEN-GREEN INTERARRIVAL TIME
REPEAT 12.
002306 000000 000000 Z 04180 DCPI: ZBLOK 12; * COMPUTE TIME PER INTERACTION
REPEAT 12.
002320 000000 000000 Z 04190
04200 ; LINEAR BUCKETS
04210 ; RANGE: >=5,5-10,10-15, ... ,40-45,45-78
04220 ;
04230 ; CPIL: ZBLOK 12; CHAR/INPUT LINE
REPEAT 12.
002332 000000 000000 Z 04240 CPOL: ZBLOK 12; CHAR/OUTPUT LINE
REPEAT 12.
Z 002344 000000 000000

```

04260      ;          CHARACTER (MOSTLY) INTERRUPTS/SECOND/USER
04270      ,          RANGE: >=15.15-7.5,7.5-3,3-1.5, ... ,0.075-.03,<.03
04280
04290
04300      LINT: ZBLOK 12; INPUT INTERRUPTS
              REPEAT 12,
              Z
002356    000000 000000 04310      OINT: ZBLOK 12; OUTPUT INTERRUPTS
              REPEAT 12,
              Z
002370    000000 000000 04320      TINT: ZBLOK 12; TOTAL INTERRUPTS
              REPEAT 12,
              Z
002402    000000 000000 04330      T5: ZBLOK 12; INTERPRETATION RATE
              REPEAT 12,
              Z
002414    000000 000000 04340      T6: ZBLOK 12; BLOCKSIZE DISTRIBUTION
              REPEAT 12,
              Z
002426    000000 000000 04350      T7: ZBLOK 12; MISC COUNTS
              REPEAT 12,
              Z
002440    000000 000000 04360      T7..9=T7+9 ZBLOK 12; TASK TURN-AROUND TIME
              REPEAT 12,
              Z
002452    000000 000000 04380      T9: ZBLOK 12; JOE USES T9, T10, AND T11 TO COUNT
              REPEAT 12,
              Z
002464    000000 000000 04390      T10: ZBLOK 12; COMMAND EXECUTIONS BY VERB TYPE
              REPEAT 12,
              Z
002476    000000 000000 04400      T11: ZBLOK 12;
              REPEAT 12,
              Z
002510    000000 000000 04410      T11..1=T11+1
04420      T11..2=T11+2
04430      T11..3=T11+3
04440      T11..4=T11+4
04450      T11..5=T11+5
04460      T11..6=T11+6
04470
04480
04490
04500      T11..7=T11+7
              T11..8=T11+8
              T11..9=T11+9

```

04520		PER USER COUNTERS
04530	,	
04540	,	RIGHT=COMPUTE TICKS SINCE LAST GREEN
04550	,	LEFT=TIME OF LAST GREEN
04560	,	
04570		
04580	USTAT:	ZBLOK N.S; USER COUNTERS
	REPEAT N.S.,	
002522 000000 000000	Z	
04590	MINT:	ZBLOK N.S; CURRENT USERS INITIALS
04600	REPEAT N.S.,	
002572 000000 000000	Z	
04610	LIST	
04620	SUM:	ZBLOK N.S; USER BLOCK CHECKSUMS
	REPEAT N.S.,	
002642 000000 000000	Z	

```

04640      IN CTP MACRO: A=# OF COLUMNS (MUST TOTAL <=72)
04650      ;
04660      B=SCALE FACTOR USED BEFORE PRINT
04670      C=CT (OR COUNTER) #
04680      E=0 IF COUNTER IS ZEROED EACH MINUTE
04690      DEFINE CTP [A,B,C,E]
<OPDEF CTP2 [BYTE (1) E (5) A (12) +D *B]
04700      CTP2 CT* C>
04710      CTPAR: CTP 3,1,32,1; UR - USERS
OPDEF CTP2 [BYTE (1) 1(5) 3 (12) +D1]
CTP2 CT32
04720      CTP 3,1,33; GQ - GREEN USERS
OPDEF CTP2 [BYTE (1) (5) 3 (12) +D1]
CTP2 CT33
04730      CTP 2,1,34; C - COMPUTING USERS
OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1]
CTP2 CT34
04740      CTP 2,1,35; B - OUTPUT LIMITED USERS
OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1]
CTP2 CT35
04750      CTP 4,6,13; COM- COMPUTE TIME
OPDEF CTP2 [BYTE (1) (5) 4 (12) +D6]
CTP2 CT13
04760      CTP 4,10,14; STA - STATEMENTS INTERPRETED
OPDEF CTP2 [BYTE (1) (5) 4 (12) +D10]
CTP2 CT14
04770      CTP 2,1,15,1; A - IGNORED PARITY ERRORS
OPDEF CTP2 [BYTE (1) 1(5) 2 (12) +D1]
CTP2 CT15
04780      CTP 2,1,50; M - CORE COMPACTS
OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1]
CTP2 CT50
04790      CTP 4,1,16; TL - LINES TRANSMITTED TO USERS
OPDEF CTP2 [BYTE (1) (5) 4 (12) +D1]
CTP2 CT16
04800      CTP 3,1,31; CR - LINES RECEIVED FROM USERS
OPDEF CTP2 [BYTE (1) (5) 3 (12) +D1]
CTP2 CT31
04810      CTP 3,10,25; CI - CHARACTERS INPUT
OPDEF CTP2 [BYTE (1) (5) 3 (12) +D10]
CTP2 CT25
04820      CTP 3,100,26; CO - CHARACTERS OUTPUT
OPDEF CTP2 [BYTE (1) (5) 3 (12) +D100]
CTP2 CT26
04830      CTP 2,1,41,1; T - TAPE ERRORS
OPDEF CTP2 [BYTE (1) 1(5) 2 (12) +D1]
CTP2 CT41
04840      CTP 2,1,22A,1; K - DISC ERRORS
OPDEF CTP2 [BYTE (1) 1(5) 2 (12) +D1]
CTP2 CT22A
002712 430001 002160*
002713 030001 002161*
002715 020001 002163*
002716 040006 002165*
002717 040012 002166*
002720 420001 002167*
002721 020001 002230*
002722 040001 002171*
002723 030001 002173*
002724 030012 002174*
002725 030144 002175*
002726 420001 002221*
002727 420001 002225*

```

		CTP 2,1,27A;	P - LAST STATION TO TRANSMIT PARITY ERROR
002730	020001 002176*	OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT27A	
002731	020001 002226*	04870 CTP 2,1,22; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT22	C - CONSOLE PARITY ERRORS
002732	020001 002177*	04880 CTP 2,1,27B; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT27B	B - LAST STATION TO TRANSMIT A BAD CHARACTER
002733	020001 002227*	04890 CTP 2,1,24; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT24	S - SCANNER PARITY ERRORS
002734	420001 002224*	04900 CTP 2,1,21; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT21	D - DRUM ERRORS
002735	430144 002212*	04910 CTP 3,100,43,1; OPDEF CTP2 [BYTE (1) (5) 3 (12) +D100] CTP2 CT43	1/100% COMPUTE TIME - RESET EACH HOUR (LOGS AS %)
002736	033720 002201*	04920 CTP 3,2000,11; OPDEF CTP2 [BYTE (1) (5) 3 (12) +D2000] CTP2 CT11	ID - IDLE LOOP COUNT
002737	020001 002200*	04930 CTP 2,1,12; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT12	U - UNOVERLAPPED I/O COUNTS
002740	430001 002202*	04940 CTP 3,1,39,1; OPDEF CTP2 [BYTE (1) (5) 3 (12) +D1] CTP2 CT39	OD - USERS IN CORE
002741	040001 002203*	04950 CTP 4,1,19A; OPDEF CTP2 [BYTE (1) (5) 4 (12) +D1] CTP2 CT19A	SW - DRUM READS+WRITES
002742	020001 002204*	04960 CTP 2,1,18; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT18	S - DISC SAVES
002743	020001 002207*	04970 CTP 2,1,19; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT19	L - DISC LOADS
002744	020001 002205*	04980 CTP 2,1,18A; OPDEF CTP2 [BYTE (1) (5) 2 (12) +D1] CTP2 CT18A	D - DISC DISCARDS
		04990 CTPARS=-. -CTPAR	

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
DATA CONTROL INTERRUPT ROUTINE

PAGE 22

002745	000000	000000	05010	DISMIS: Z;	CONO	DATA CONTROL END ROUTINE
002746	700600	001100	05020		JRST	TURN OFF CHANNEL
002747	254520	002745*	05030			PI,1000+CHDC;
			05040			12,@DISMIS

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
DRUM CHANNEL INTERRUPT ROUTINES

PAGE 23

002750	000000	000000	05060	DRMR:	Z	SETZM	DRM14;	IF ERROR, FLAG STORED HERE
002751	402000	003110*	05070			CONSZ	DP,100060;	MISS DATA, NO EX MEM, OR PARITY ERROR
002752	701300	100060	05080			JRST	DR,	DEER;
002753	254000	003126*	05090			SKIPN	DBXY	SHOULD BE USING THE DRUM
002754	336000	002107*	05100			HALT	15;	
002755	040000	000015	05110	H15:		CONSO	DP,10	
002756	701340	000010	05120			JRST	DRM3	KILL THE INTERRUPT
002757	254000	002762*	05130			CONO	DP,0;	DISMIS END INTERRUPT FROM 167
002760	701200	000000	05140			JRST	12,ADRMR;	
002761	254520	002750*	05160					
002762	250040	003103*	05170	DRM3:	EXCH	B,DRM9		
002763	250100	003102*	05180		EXCH	C,DRM8		CHECK FOR ERROR
002764	740300	001000	05190		CONSZ	DR,1000;		
002765	254000	003133*	05200		JRST	DERR		
002766	201040	077777	05220		MOVEI	B,77777		MASK OUT POSSIBLE HIGH CARRY BIT
002767	406040	003326*	05230		ANDM	B,DPWD1;		
002770	701040	000001	05240		DATAI	DP,B		
002771	312040	003326*	05250		CAME	B,DPWD1;		CHECK FOR GOOD
002772	254000	003133*	05260		JRST	DERR		
002773	740040	000001	05270		DATAI	DR,B		DRUM REGISTER ENDING CONTENTS
002774	312040	003165*	05280		CAME	B,DEDR1;		
002775	254000	003133*	05290		JRST	DRERR		
002776	740340	000100	05300	H17:	HALT	DR,100		
002777	040000	000017	05310		CONO	DR,270;		NO JOB DONE FLAG
003000	740200	000270	05320		SETZM	DBXY		DESSELECT THE DRUM
003001	402000	002107*	05330		SKIPN	DMWR;		
003002	336000	003330*	05340		JRST	10,DRM7;		SKIP IF WRITING
003003	254400	003043*	05350		SETZM	DEC'T;		MUST BE A READ
003004	402000	003163*	05360		AOS	B		ZERO THE ERROR COUNT
			05370					
			05380					
003005	250700	003107*	05390		EXCH	S,DRM13		
003006	200700	003327*	05400		MOVE	S,DMUSR		
003007	200100	003332*	05410		MOVE	C,BLKSWRIT		
003010	200040	003324*	05420		MOVE	B,DMBK		
003011	402001	000272*	05430		SETZM	CORE (B);		IDLE THE CORE JUST WRIT
003012	350000	000001	05440		AOS			
003013	367100	003011*	05450		SOJG	C,*-2		
003014	350000	000320*	05460	DR2:	AOS	N,DRM;		COUNT DRUM USERS
003015	370000	002202*	05470		SOS	CT39;		COUNT DOWN USERS IN CORE
003016	200040	003305*	05480		MOVE	B,DMNR;		GET LIST FOR OUT
003017	603040	777777	05490		TLNE	B,777777;		SKIP IF NO MORE TO GO OUT
003020	254400	003113*	05500		JRST	10,DM10;		GO FOR NEXT WRITE
003021	332000	005470*	05510		SKIPE	SS98;		SKIP IF NO COMPACT REQUEST
003022	254400	003037*	05520		JRST	10,DRM2		
003023	335700	003331*	05530		SKIPGE	S,DMIN;		SKIP IF SOME TO COME IN
003024	254400	003031*	05540		JRST	10,DRM4.5		
003025	250740	003111*	05550		EXCH	PP,DRMPP		
003026	260740	003333*	05560		TSX	ISWAP;		
003027	250740	003111*	05570		EXCH	PP,DRMPP		
003030	254400	003032*	05580		JRST	10,DRM5		

```

05610      ;   INTERRUPT END SEQUENCE
05620
05630      DRM4.5: AOS      COMEBACK
05640      DRMS:   EXCH     S, DRM13
05650      DRM6:   EXCH     B, DRM9
05660      DRM6.5: EXCH    C, DRM8
05670      AOS      CT19A;
05680      JRST     2, @DRMR
05690
05700      DRM2:   AOS      SS99;
05710      SETZM   SS98
05720      AOS      DMBY
05730      JRST     DRM4.5
05740

```

	003031	003032	003033	003034	003035	003036	003037	003040	003041	003042
3500000	2500700	2500400	2501000	3500000	2541200	3500000	3500000	4020000	3500000	2540000
002111*	003107*	003103*	003102*	002203*	002750*	005471*	005470*	002107*	003031*	003030

COUNT DRUM ACTIONS

			READ	END	INTERRUPT
003043	250700	0033331*	05760	;	
003044	250000	003110*	05770	;	
003045	550040	0033324*	05780:	DRM7:	EXCH S,DMIN
003046	271040	000020	05790		EXCH A,DRM14
003047	135100	000264*	05800		HRRZ B,DMIBK
003050	135000	000261*	05810		ADDI B,BBLOCK
003051	306000	000006	05820		LDB C,S-BLOCK
003052	275100	000001	05830		LDB A,S,STA
003053	264000	003207*	05840		CAIN A,QC,S;
003054	316016	002642*	05850		SUBI C,1
003055	254000	003061*	05860		JSR CKSUM;
003056	250000	003110*	05870		CAMN A,SUM(S)
003057	250700	0033331*	05880		JRST DRM7-5;
003060	254000	003137*	05890		EXCH A,DRM14
003061	402000	003163*	05900		EXCH S,DMIN
003062	250000	003110*	05910		JRST DMERR
003063	550040	003324*	05920		SETZM DECT
003064	137700	000317*	05930		EXCH A,DRM14
003065	271040	000020	05940		HRRZ B,DMIBK
003066	137040	000263*	05950		DBP S,S,UR
003067	250000	003105*	05960		ADDI B,BBLOCK
003070	550040	003324*	05970		DBP B,S-COR;
003071	135100	000264*	05980		EXCH A,DRM11;
003072	137000	000316*	05990		HRRZ B,DMIBK
003073	350000	000001	06000		LDB C,S-BLOCK
003074	367100	003072*	06010		DBP A,S,UD;
003075	250000	003105*	06020		AOS B
003076	350000	002202*	06030		SOJG C,-2,
003077	370000	000320*	06040		EXCH A,DRM11
003100	250700	003331*	06050		AOS CT39
003101	254000	003033*	06060		SOS N,DRM
			06070		EXCH S,DMIN
			06080		DRM6 JRST
			06090		

003102	000000	000000	06110	DRM8:	Z;	SAVE FOR REG C
003103	000000	000000	06120	DRM9:	Z;	SAVE FOR REG B
003104	000000	000001	06130	DRM10:	OCT	SAVE FOR REG A AND A ONE
003105	000000	000002	06140	DRM11:	OCT	SAVE FOR REG A AND A TWO
003106	000000	000000	06150	DRM12:	Z	SAVE CELL FOR S
003107	000000	000000	06160	DRM13:	Z;	SAVE FOR A
003110	000000	000000	06170	DRM14:	Z;	LOCAL PUSH REGISTER
003111	77776	003111	06180	DRMPP:	XWD	-2..;
003112	000000	000000	06190		Z	
			06200			
			06210			
			06220			
			06230			INITIATE NEXT WRITE
			06240			
			06250			B HAS PP REG FOR REQUEST LIST
			06260			
003113	250740	003111	06270	DM10:	EXCH	PP, DRMPP
003114	262040	000016	06280		POP	B, S;
003115	135100	000264	06290		LDB	C, S, BLOCK
003116	250140	003106	06300		EXCH	D, DRM12
003117	202040	003305	06310		MOVE	B, DMNR
003120	250000	003110	06320		EXCH	A, DRM14
003121	260740	003234	06330		TSX	OSWAP;
003122	250000	003110	06340		EXCH	A, DRM14
003123	250140	003106	06350		EXCH	D, DRM12
003124	250740	003111	06360		EXCH	PP, DRMPP
003125	254000	003032	06370		JRST	DRM5

		06390	;	DATA PROCESSOR ERRORS (PARITY, NOEX MEM, DATA MISS)
003126	250040 003103*	06400		
003127	701240 000001	06410	DEER:	EXCH B,DRM9
003130	506040 003110*	06420		CONI DE,B
003131	701200 000000	06430		HRLM B,DRM14
003132	250100 003102*	06440		CONO DP,0
		06450		EXCH C,DRMB
		06460	;	OTHER HARDWARE ERRORS--DC FLAGS AND DATA REGISTER MISCOMPARE
003133	701240 000001	06480	DERR:	CONI DR,B
003134	542040 003110*	06490		HRRM SAVE FLAGS FOR DISPLAY
003135	740200 002270	06500		CONO DR,200270;
003136	254400 003137*	06510		JRST DISMISS DRUM
003137	350000 002224*	06520		
003140	350040 003163*	06530	DMERR:	AOS CT21;
003141	303040 000005	06540		AOS B,DECT
003142	254000 003166*	06550		CAILE B,5
003143	350000 .002107*	06560		JRST FAIL
003144	336000 003330*	06570		AOS DMBY COUNT DRUM ERRORS
003145	254000 003154*	06580		DMWR; JRST
003146	701200 000102	06590		DE10; B,5
003147	701140 003325*	06600		CAILE DR,100+CDRM
003150	740140 003164*	06610		JRST FAIL
003151	740200 000260	06620		AOS SKIP IF WRITE
003152	740200 000222	06630		CONO GO RECOVER READ ERROR
003153	254000 003033*	06640		DATAO DP,DFWD
		06650		DATAO DP,DFWD
		06660		DATAO DR,DEDR
		06670		CONO DR,260
003154	701200 000002	06680	DE10:	CONO DR,220+CDRM
003155	701140 003325*	06690		JRST DRM6
003156	740140 003164*	06700		
003157	740200 000260	06710		
003160	740200 000232	06720		
003161	254000 003033*	06730		
		06740		
003162	000000 000000	06750	CKER:	SET IF A GRONKED USER
003163	000000 000000	06760	DECT:	ERROR RETRY COUNT
003164	000000 000000	06770	DEDR:	UNIT AND TRACK OF LAST SELECT
003165	000000 000000	06780	DEDR1:	EXPECTED ENDING TRACK AND SECTOR

06800 ; CANT RECOVER DRUM ERROR--PREPARE TO GRONK USER
06810
003166 250700 0033331*
003167 250000 003110*
003170 202016 002572*
003171 350000 003162*
003172 201000 000001
003173 137000 000270*
003174 250000 003110*
003175 402000 003163*
003176 402000 002107*
003177 332000 003330*
003200 254000 003100*
003201 135100 000264*
003202 550040 003324*
003203 402001 000272*
003204 350000 000001
003205 367100 003203*
003206 254000 003077*
FAIL:
EXCH S, DMIN
EXCH A, DRM14
MOVEM A, MINT (S) ;
AOS CKER;
MOVEI A, 1
DPB A, S, GK;
EXCH A, DRM14
SETZM DECT
SETZM DMBY
SKIPE DMWR
JRST DRM7.7
LDB C,S,BLOCK
HRRZ B,DMIBK
SETZM CORE (B) ;
AOS B
SOJG C,-2
JRST DRM7.6

REPORTS ZERO CHECKSUM WHEN CKF IS ZERO C, B CLOBBERED S= USER #, C=BLOCKS TO SUM, B= LOCATION			
003207	000000 000000	07000	
003210	332000 002114*	07010	
003211	254000 003214*	07020	
003212	201000 000000	07030	CKSUM: 2
003213	254120 003207*	07040	SKIPE CKF
003214	240100 000012	07050	JRST *+3
003215	275100 000001	07060	MOVEI A,0
003216	240040 000012	07070	JRST 2,ACKSUM
003217	542040 003231*	07080	ASH C,1 D10
003220	250340 003231*	07090	SUBI C,1
003221	250400 003232*	07100	ASH B,1 D10
003222	250440 003233*	07110	HRRM B,CKS7
003223	201000 000000	07120	EXCH 7,CKS7
003224	254000 000007	07130	EXCH 10,CKS10
003225	250340 003231*	07140	EXCH 11,CKS11
003226	250400 003232*	07150	MOVEI A,0
003227	250440 003233*	07160	JRST 7
003230	254120 003207*	07170	CKS5: EXCH 7,CKS7
		07180	EXCH 10,CKS10
		07190	EXCH 11,CKS11
		07200	JRST 2,ACKSUM
		07210	
		07220	
003231	270002 000000	07230	CKS7: ADD A,0 (Q)
003232	365100 000007	07240	CKS10: SOJGE C,7
003233	254000 003225*	07250	CKS11: JRST CKS5
		07260	
		07270	
		07280	

		C= # OF BLOCKS TO WRITE S= USER TO WRITE DRUM LOCATIONS ARE FIXED BY USER - TWO TRACKS EACH, SO THE MAX SIZE HANDLED IS 16K WORDS PER USER. B,D ARE CLOBBERED
07300	07350	O SWAP: CONSZ DR, 200
07310	H14:	HALT 14; DRUM SHOULD BE FREE
07320		MOVEM C,BLKSWRIT
07330		LDB B,S,COR
07340		JSR CKSUM
003234	740300 000200	MOVEM A,SUM(S)
003235	040000 000014	MOVE C,BLKSWRIT
003236	202100 003332*	ASH C,6
003237	135040 000263*	MOVEM C,DEDRI
003240	264000 003207*	ASH C,4
003241	202016 002642*	MOVEM C,DPWD1
003242	200100 003332*	MOVE C,BLKSWRIT
003243	240100 000006	LSH D,C
003244	202100 003165*	MOVEM C,DPWD;
003245	240100 000004	MOVEM B,BBLOCK
003246	202100 003326*	MOVEM B,DMIBK;
003247	200100 003332*	MOVEM C,1
003250	550140 000002	MOVEI C,S.ID;
003251	242100 000034	DPB
003252	212100 003325*	WORDS TO MOVE
003253	135040 000263*	LSH
003254	275040 000020	MOVEM B,S,COR
003255	202040 003324*	MOVEM B,BBLOCK
003256	201100 000001	MOVEM B,DMIBK;
003257	137100 000315*	MOVEI C,1
003260	350000 000001	SET DRUM USE BIT
003261	367140 003257*	AOS B
003262	135040 000263*	SOJG D,-2
003263	240040 000012	LDB B,S,COR
003264	137040 000263*	ASH B,1D10
003265	542040 003325*	DPB B,S,COR;
003266	272040 003326*	HRM B,DPWD;
003267	701200 000102	ADDM B,DPWD1;
003270	701140 003325*	CONO DP,100+CDRM;
003271	550040 000016	DATAO DP,DPWD;
003272	552040 003327*	HRRZ B,S
003273	270040 000321*	HRRZM B,DMUSR
003274	240040 000012	ADD B,DBASE
003275	202040 003164*	ASH B,1D10;
003276	740140 000001	MOVEM B,DEDRI
003277	740200 000260	DATAO DR,B;
003300	740200 000222	CONO DR,260;
003301	272040 003165*	ADDM B,DEDRI
003302	350000 003330*	AOS DMWR;
003303	350000 002107*	INDICATE DRUM WRITE
003304	263740 000000	AOS DMBY
		DONE

003305 000000 003305* 07780 DMNR: XWD 0...:
07790 ZBLOK 16

REPEAT 16,

003306 000000 000000	Z;	DMIBK:	Z;	INCOMING BLOCK NUMBER
003324 000000 000000	07800	DPMD:	Z;	I/O CONTROL WORD
003325 000000 000000	07810	DPMID:	Z;	EXPECTED END CONTENTS OF 167
003326 000000 000000	07820	DMUSR:	Z;	DRUM USER
003327 000000 000000	07830	DMVR:	Z;	SET WHEN WRITING
003330 000000 000000	07840	DMIN:	Z;	RIGHT = USER TAGGED FOR IN, LEFT = PLACE IN QUEUE, NEG IF NO IN
003331 000000 000000	07850	BLKSWR:	Z;	* OF BLOCKS WRITTEN
003332 000000 000000	07860			
	07870			
	07880			

07900		S HAS USER TO COME IN; DMIBK= CORE BLOCK FOR "IN"	
07910		S IS CLOBBERED	
07920	I\$WAP:	LDB C,S,BLOCK LDB B,S,STA CAIN B,Q,C,S; SUBI C,1 MOVEM S,DMIN MOVEM C,S; LSH C,1D28 MOVNM C,DPWD; HRRZ B,DMIBK; ADDI B,BBLOCK ASH B,1D10 B,DPWD; FIRST WORD	
		CHECK FOR SIZE INCREASE	
0033333	135100 000264*	07930	
0033344	135040 000261*	07940	
0033355	306040 000006	07950	
0033366	275100 000001	07960	
0033377	202700 003331*	07970	
0033400	202100 000016	07980	
0033411	242100 000034	07990	
0033422	212100 003325*	08000	
0033433	550040 003324*	08010	
0033444	271040 000020	08020	
0033455	240040 000012	08030	
0033466	542040 003325*	08040	
0033477	552040 003326*	08050	
0033500	550040 003324*	08060	
0033511	200100 000016	08070	
0033522	240100 000012	08080	
0033533	272100 003326*	08090	
0033544	200100 000016	08100	
0033555	240100 000006	08110	
0033566	202100 003165*	08120	
0033577	200100 000016	08130	
0033600	250000 003104*	08140	
0033611	137000 000315*	08150	
0033622	350000 000001	08160	
0033633	367100 003361*	08170	
0033644	402000 003330*	08180	
0033655	350000 002107*	08190	
0033666	200700 003331*	08200	
0033677	270700 000321*	08210	
0033700	240700 000012	08220	
0033711	701200 000002	08230	
0033722	701140 003325*	08240	
0033733	202700 003164*	08250	
0033744	700600 000400	08260	
0033755	740140 000016	08270	
0033766	740200 000260	08280	
0033777	740200 000232	08290	
0034000	250000 003104*	08300	
0034011	700600 000200	08310	
0034022	272700 003165*	08320	
0034033	553000 003165*	08330	
0034044	263740 000000	08340	
		08350	

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
TAPE INTERRUPT ROUTINES

PAGE 33

003405	722640	003444*	08370	TIP:	CONI	MT1,TEND4;	READ IN TAPE FLAGS
003406	722740	000004	08380		CONSO	MT1,4;	ERF INTERRUPT?
003407	254000	003432*	08390		JRST	TE20	
003410	722600	000000	08410	TEND:	CONO	MT1,0;	DISABLE THE INTERRUPTS
003411	250700	003445*	08420		EXCH	S,TEND5	
003412	250040	003446*	08430		EXCH	B,TEND6	
003413	250200	003447*	08440		EXCH	E,TEND7;	GIVE ME SOME ELBOW ROOM
003414	250740	003530*	08450		EXCH	PP,TPDL	
003415	200040	003444*	08460		MOVE	B,TEND4;	GET THE FLAGS
003416	602040	000070	08470		TRNE	B,70;	SKIP IF NO ERROR
003417	254000	003451*	08480		JRST	TEND9;	GO FIX THE ERROR
003420	402000	003450*	08490		SETZM	TEND8;	OK, ZERO THE RETRY COUNT
003421	332000	003444*	08500		SKIPE	TEND4;	RESTART SO DONT GIVE BACK BUFFER
003422	260740	010365*	08510		TSX	MBA	
003423	332000	000327*	08520		SKIPE	DT,BUF;	SKIP IF NO MORE BUFFERS
003424	260740	003475*	08530	TEND1:	TSX	CIO	
003425	250700	003445*	08540	TEND2:	EXCH	S,TEND5	
003426	250040	003446*	08550		EXCH	B,TEND6	
003427	250200	003447*	08560		EXCH	E,TEND7	
003430	250740	003530*	08570		EXCH	PP,TPDL	
003431	254520	003544*	08580		JRST	12,ACTYR	
003432	336000	003450*	08590		SKIPI	TEND8;	SKIP TO IGNORE BKSPACE OR WRITE BLANK
003433	254000	003436*	08600	TE20:	JRST	TE22;	GO TO TRY TO READY TAPE
003434	722600	000000	08610		CONO	MT1,0;	KILL THE INTERRUPT
003435	254520	003544*	08620	TE21:	JRST	12,ACTYR	
003436	700700	000100	08630		CONSZ	PI,CHDC;	GET IT LATER IF DATA CONTROL IS BUSY
003437	254000	003434*	08640	TE22:	JRST	TE21	
003440	722740	000002	08650		CONSO	MT1,2;	SKIP IF TRANSPORT READY
003441	254000	003434*	08660		JRST	TE21;	RETURN WE WILL TRY AGAIN LATER
003442	402000	003444*	08670		SETZM	TEND4;	SET RESTART SIGNAL
003443	254000	003410*	08680		JRST	TEND;	GO DO IT
003444	000000	000000	08700				
003445	777777	777776	08710	TEND4:	Z;		FLAG STORAGE
003446	000000	000000	08720	TEND5:	DEC	-2;	SAVE CELL FOR S AND CONTEXT FOR TAPE
003447	000000	000000	08730	TEND6:	Z		SAVE CELL
003450	000000	000000	08740	TEND7:	Z;		COUNT OF REWRITE TRIES
			08750	TEND8:	Z;		
			08760				

```

003451 350000 002221*   08780   TEND9: AOS      TERROR;
003452 350040 003450*   08790   TEND9: AOS      COUNT REWRITE TRIES
003453 722600 000000   08800   CONO      MT1,0;
003454 260740 003470*   08810   TSX       OFF WITH THE LIGHTS
003455 722200 003406   08820   CONO      WAIT ON TAPE
003456 260740 003470*   08830   TSX       MTC,CDAT+3400;
003457 307040 000003   08840   CAIG      BACKSPACE TAPE
003460 254000 003424*   08850   JRST      TRY REWRITE 3 TIMES.
                                                TEND 1

003461 722200 005406   08860   TEND95: CONO     MTC,CDAT+5400;
003462 260740 003470*   08870   TSX       WRITE BLANK TAPE
003463 722700 000030   08880   CONSZ    MT1,30;
003464 350000 002223*   08890   AOS      CT45;
003465 350000 002222*   08900   AOS      TERASE
003466 402000 003450*   08910   SETZM   TEND 8
003467 254000 003424*   08920   JRST      TEND1
                                                TEND 1

003470 722740 040000   08930   TWAIT: CONSO    MT1,40000;
003471 254000 003470*   08940   JRST      -1
003472 722740 000002   08950   CONSO    MT1,2;
003473 254000 003472*   08960   JRST      AND FOR TAPE READY
003474 263740 000000   08970   08980
                                08990
                                09000
                                09010

```

			MAG TAPE OUTPUT START ROUTINE		
09030	;	CIO:	SKIPE	SKR;	SKIP IF SKULLER NOT USING DC
09040			JRST	CIO1	NO START IF DISC USING DC
09050			SKIPN	DIP;	
09060			CONSZ	MT1,200	CANT WRITE ON FILE PROTECTED TAPE
09070			JRST	CIO1;	
09080			CONSO	MT1,2;	TAPE SHOULD BE READY
09090			JRST	CIO1;	TAPE MUST BE READY
09100			MOVE	B,[BLKO DC,TCW]	
09110			MOVEM	B,40*2*CDC;	SET UP DC INTERRUPT ROUTINE
09120			MOVE	B,[USR DISMIS]	
09130			MOVEM	B,41*2*CDC	
09140			HRRZ	B,DT.BUF;	ADDRESS OF BUFFER
09150			HRRM	B,TCW	
09160			HLRZ	B,1(B);	GET WORD COUNT
09170			MOVNS	B,B	
09180			HRLM	B,TCW;	NEG COUNT TO CONTROL WORD
09190			CONO	MT1,0;	DISABLE ALL, I DONT KNOW WHY!
09200			CONO	DC,CDC+DMT+3400;	START DATA OUT
09210			CONO	MTC,CDAT+71000;	START TAPE - BINARY @556 BPI
09220			CONO	PI,2000+CHDC;	ENABLE DATA CONTROL
09230			SETZM	TAPES	
09240			CONO	MT1,1;	ENABLE TAPE CONTROL FREE TRAP
09250			DONE		
09260					
09270					
09280					
09290		CIO1:	AOS;		SET RESTART FLAG
09300		DONE			
09310					
09320			TAPES:	Z;	SET IF TAPE RESTART IN NEEDED
09330			TCW:	Z;	DC CONTROL WORD FOR TAPE
09340			TPDL:	-2,-;	PUSH LIST DURING INTERRUPT
09350			Z		
09360					
09370					
09380					
09390			TRST:	SKIPN	NO RESTART IF DISC USING DC
09400				TAPES	NOR IF NONE INDICATED
09410				DONE;	SKIP IF NOT BUSY
09420				CONSZ	
09430				TTV,170;	
09440				JRST	TRST1
09450				MOVEI	B,"";
09460				DATAO	THATS A BELL
09470				SKIPE	SKIP IF NOTHING TO START
09480				DT.BUF;	START TAPE I/O
09490				TSX	
				DONE	

		CHANNEL 6 INTERRUPTS: TTY IN, TTY OUT, AND MAG TAPE	
09510	;	CTYR:	Z
09520		CONSO	TTY,10;
09530		JRST	ST41;
09540		SKIN	CO.BUF;
09550		JRST	ST40;
09560		EXCH	B,ST31;
09570		ILDB	B,ST32;
09580		JUMPE	B,ST33;
09590		DATAO	TTY,B;
09600		EXCH	B,ST31;
09610		JRST	12,ACTYR;
09620		CONO	TTY,200+CCTY;
09630		EXCH	S,ST35;
09640		EXCH	PP,ST22
09650		EXCH	E,ST24
09660		TSX	MBA
09670		SKINE	CO.BUF;
09680		TSX	BIO
09690		EXCH	PP,ST22
09700		EXCH	E,ST24
09710		EXCH	S,ST35
09720		JRST	ST34
09730		CONO	TTY,200+CCTY
09740		EXCH	12,ACTYR
09750		ST40:	DATAO
09760		JRST	12,ACTYR
09770		XWD	-2,-
09780		ST22:	PDL FOR INTERRUPT ROUTINE
09790		ST23:	
09800		ST24:	
09810		ST31:	
09820		ST32:	POINT 7,0;
09830		ST35:	-1
09840			DEC
09850			START ROUTINE FOR CONSOLE TTY OUTPUT
09860		BIO:	HRRZ
09870			B,CO.BUF
09880			B,IP POINT 7,0
09890		HLL	ADDI B,1;
09900			MOVEM B,ST32;
09910			CONO TTY,CCTY;
09920			ILDB B,ST32
09930			DATAO TTY,B;
09940			DONE
09950			09960
09970			

```

09990 ; INPUT INTERRUPT FROM CONSOLE TTY
10000
003612 712340 000040 10010 CONSO TTY,40; SKIP IF TTY INPUT INTERRUPT
003613 254000 003405* 10020 JRST TIP;
003614 250040 003664* 10030 EXCH B,ST46
003615 250200 003665* 10040 EXCH E,ST47
003616 250740 003672* 10050 EXCH PP,ST52
003617 326200 003627* 10060 JUMPN E,ST42;
003620 260740 010343* 10070 TSX GETBUF
003621 254000 003657* 10080 JRST ST45-1;
003622 201040 000120 10090 MOVEI B,1DB0;
003623 202040 003671* 10100 MOVEM B,ST51
003624 542200 003666* 10110 HRRM E,ST48;
003625 500200 010535* 10120 HLL E, [POINT 7,0];
003626 350000 000004 10130 AOS E; MAKE UP POINTER
003627 712040 000001 10140 DATAI TEXT IS WORD ONE
003630 136040 000004 10150 IDPB B,E;
003631 376000 003671* 10160 SOSN ST51;
003632 254000 003635* 10170 JRST .+3;
003633 302040 000215 10180 CAIE B,215;
003634 254000 003660* 10190 JRST ST45;
003635 250140 003667* 10210 EXCH D,ST49
003636 201040 000000 10220 MOVEI B,0
003637 200200 003666* 10230 MOVE E,ST48; GET BUFFER LOCATION
003640 350000 000004 10240 AOS E;
003641 202200 003670* 10250 MOVEM E,ST50; TO TEXT
003642 200140 010536* 10260 MOVE D, [POINT 7,OPMSG] SAVE IT
003643 134200 003670* 10270
003644 306200 000015 10280 ST43: ILDB E,ST50
003645 254000 003652* 10290 CAIN E,15; SKIP IF NOT CARRIAGE RETURN
003646 136200 000003 10300 JRST ST44; CR - END OF LINE
003647 350000 000001 10310 IDPB E,D;
003650 305040 000036 10320 AOS B PUT CHARACTER IN MESSAGE
003651 254000 003643* 10330 CAIGE B,1D30;
003652 202040 004562* 10340 JRST ST43; SKIP IF ENOUGH CHAR FOR MESSAGE LINE
003653 250140 003667* 10350 MOVEM B,L,OPM; COUNT OF CHARACTERS IN MESSAGE
003654 550200 003666* 10370 EXCH D,ST49
003655 260740 010401* 10380 HRRZ E,ST48;
003656 201200 000000 10390 TSX MBA1; GET BUFFER POINTER
003657 712200 001006 10400 MOVEI E,0; AND GIVE IT BACK
003660 250200 003665* 10410 CONO TTY,CCTY+1000;
003661 250040 003664* 10420 EXCH E,ST47
003662 250740 003672* 10430 EXCH B,ST46
003663 254520 003544* 10440 EXCH PP,ST52
003664 000000 000000 10450 JRST 12,ACTYR
003665 000000 000000 10460
003666 440700 000000 10470 ST46: Z;
003667 000000 000000 10480 ST47: Z;
003668 000000 000000 10490 ST48: POINT 7,0;
003669 000000 000000 10500 ST49: Z;
003670 000000 000000 10510 ST50: Z;
003671 000000 000000 10520 ST51: Z;
003672 777776 003670* 10530 ST52: XWD -2,ST51-1;

```

SAVE REG B
BYTE POINTER FOR INPUT,SAVE REG E, SIGNAL
CONSTANT IN LEFT, BUFFER POINTER IN RIGHT
SAVE REG D
HOLDS BYTE POINTER
COUNT OF INPUT CHARACTERS
PDL POINTER FOR INTERRUPT ROUTINE

USES 00X CLASS OF PROGRAMMED OPERATORS. ASSUMES AN OUTPUT
POINTER IN C, CLOBBERS A AND B. CHARACTERS SHOULD BE ASCII
AND MUST BE LEFT JUSTIFIED IN THE "FROM" LOCATION.

```

10560      ;
10570      ;
10580      ;
10590      ;

003673    000000 00000000      PROP:      Z
003674    250040 000030      10610      EXCH      B,30;
003675    135040 010537      10620      LDB,[POINT 3,40,5]; GET OP CODE
003676    322040 003716      10630      JUMPE    B,XMTR;
003677    307040 000004      10640      CAIG     B,4;
003700    254001 003701      10650      JRST     .+1(B);
003701    254000 003706      10660      JRST     PR1;
003702    254000 003727      10670      INSR;
003703    254000 003732      10680      JRST     R2;
003704    254000 003735      10690      JRST     R3;
003705    254000 003740      10700      JRST     R4;

003706    550440 003673      10710      PR1:      HRZ      J,PROP
003707    307440 000000      10720      CAIG     J,INTBEG
003710    254000 003713      10730      JRST     H3;
003711    201000 000075      10740      MOVEI   A,75
003712    254000 004162      10750      JRST     KILL;
003713    201040 000003      10760      JRST     GO TO ELIMINATE THE USER
003714    202040 000040      10770      H3:      MOVEI   B,3;
003715    254000 003740      10780      MOVEM   B,40
003716    200040 010535      10790      JRST     R4;

003717    540040 000040      10800      XMTR:      MOVE    B,[POINT 7,0]
003720    202040 002112      10830      HR      B,40
003721    135040 010540      10840      MOVM   B,MISC;
003722    322040 003713      10850      LDB,[POINT 7,40,12]; GET "M" THE CHARACTER COUNT
003723    134000 002112      10860      JUMPE B,H3;
003724    136000 000002      10870      ILDB   A,MISC
003725    367040 003723      10880      IDPB   A,C;
003726    254120 003673      10890      SJG    B,-2
003727    200000 000040      10900      JRST     2,@PROP;
003730    136000 000002      10910      MOVE    A,40;
003731    254120 003673      10920      IDPB   A,C;
003732    550200 000040      10930      JRST     2,@PROP;
003733    260740 010233      10940      INSR:      MOVE    A,40;
003734    254120 003673      10950      TSX    CHS;
003735    504200 000040      10960      R2:      JRST     2,@PROP;
003736    260740 010433      10970      HRRZ E,40;
003737    254120 003673      10980      TSX    CHS;
003738    504200 000040      10990      R3:      JRST     2,@PROP;
003739    260740 010433      11000      HRL    E,40;
003740    254120 003673      11010      TSX    BJ9,1;
003741    254120 003673      11020      JRST     2,@PROP;

```

```

11040 ; HALT ROUTINE
11050 ;
11060 ;
11070 ;
11080 ;
11090 ;
11100 ;
11110 ;
11120 R4: FSW B
      MOVE B,SWITCH
      TRNN
      B,HALTS;
      JRST
      R4-1
      HRR
      B,40
      ADDI
      B,77000
      HRRM
      B,R4,99;
      MOVE
      B,30;
      JRST
      4,77000

      ; ATTEMPT RECOVERY FROM THE DRUM
11220
11230 R4..1: CONO
      CONO
      APR,210000;
      B,[XWD YEAR, 31] KILL PI SYSTEM AND PARITY FLAG
      MOVE
      BLT
      B,35; KILL PROCESSOR INTERRUPTS
      MOVE
      C,42
      MOVEM
      D,43
      MOVE
      B,0(17) SAVE DATE AND TIME
      MOVE
      B,SA+2;
      HRRZ
      B,0
      MOVE
      B,0(B)
      MOVEM
      B,SA+3;
      a0
      MOVE
      B,-1(J)
      MOVEM
      B,SA+1;
      RJD;
      JRST

003750 700600 110000
003751 700200 210000
003752 200040 010541*
003753 251040 000035
003754 202100 000042
003755 202140 000043
003756 200057 000000
003757 202040 000114
003760 550040 000000
003761 200041 000000
003762 202040 000115
003763 200051 777777
003764 202040 000113
003765 254000 000000

```

THIS ROUTINE EITHER HALTS ON ERROR OR ATTEMPTS
A RECOVERY OF THE SYSTEM FROM DRUM AND RESTART
OF JOSS FROM INITIAL CONDITIONS. APPROPRIATE ERROR
MESSAGES AND DUMPS ARE WRITTEN AT THE CONSOLE.
REGISTER 1 (B) HAS BEEN SAVED IN LOCATION 30.

CHANNEL 7 INTERRUPTS

```

11390 ; CHANNEL 7 INTERRUPTS
11400
003766 000000 000000 APRR: Z CONSO APR,1000;
003767 700340 001000 11410 JRST CHECK;
003770 254000 004141* 11420 TIME;
003771 350000 000042* 11430 SOSN D.TIME;
003772 376000 000000 11440 DISC.C;
003773 264000 000000 11450 TRY TO RECOVER DROPPED TRAP
003774 700200 001007 11460 CONO TURN OFF THE FLAG
003775 750200 000001 11470 PULSE THE "JOSS HERE" FLOP
003776 350000 000043* 11480 BUMP THE CURRENT USER CLOCK
003777 350000 000034* 11490 T.CU;
004000 350000 000044* 11500 AOS TICK;
004001 250040 000034* 11510 AOS SECONDS;
004002 305040 000074 11520 EXCH B,TICK
004003 254000 004051* 11530 CAIGE B,!D60;
004004 202100 000035* 11540 JRST CL3;
004005 350040 000033* 11550 MOVEM C,CL9;
004006 602040 000003 11560 AOS B,SEC;
004007 254000 004063* 11570 TRNE B,3;
004010 550040 003766* 11580 JRST CL6;
004011 316040 000040* 11590 CL8: HRRZ B,APPR;
004012 254000 004075* 11600 CAMN B,CL12;
004013 542040 000040* 11610 JRST CL14;
004014 402000 000041* 11620 HRRM B,CL12;
004015 200040 003740* 11630 CL18: SETZM CL13;
004016 602040 400000 11640 CL5: FSW B
004017 264000 000000 11650 MOVE B,SWITCH
004020 606040 200000 11660 TRNE B,BEEPS;
004021 254000 004025* 11670 JSR C30;
004022 200040 000033* 11680 TRNN B,SDS;
004023 307040 000005 11690 JRST CL4;
004024 264000 004017* 11700 MOVE B,SEC;
004025 ; BEEP FOR 5 SEC EACH MIN
004026 ; JSR C30

```

004025	200040	002214*	11730	CL4:	MOVE	B, CT46;	INPUT INTERRUPTS
004026	221040	003720	11740		IMULI	B, *D2000	
004027	230040	002160*	11750		IDIV	B, USERS	
004030	264000	004267*	11760		JSR	CVTL1	
004031	350002	002356*	11770		AOS	IINT (C)	
004032	200040	002215*	11780		MOVE	B, CT47;	OUTPUT INTERRUPTS
004033	221040	003720	11790		IMULI	B, *D2000	
004034	230040	002160*	11800		IDIV	B, USERS	
004035	264000	004267*	11810		JSR	CVTL1	
004036	350002	002370*	11820		AOS	OINT (C)	
004037	200040	002214*	11830		MOVE	B, CT46	
004040	270040	002215*	11840		ADD	B, CT47	
004041	221040	003720	11850		IMULI	B, *D2000	
004042	230040	002160*	11860		IDIV	B, USERS	
004043	264000	004267*	11870		JSR	CVTL1	
004044	350002	002402*	11880		AOS	TINT (C)	
004045	402000	002214*	11890	CL3:	SETZM	CT46	
004046	402000	002215*	11900		SETZM	CT47	
004047	200100	000035*	11910		MOVE	C, CL9	
004050	201040	000000	11920		MOVEI	B, 0;	ZERO THE TICKER
004051	250040	000034*	11930		EXCH	B, TICK	
004052	250040	000043*	11940		EXCH	B,T, CU	
004053	311040	000326*	11950		CAML	B,T, MAX	
004054	254000	004057*	11960		JRST	CL2;	YES TIME IS UP NO INTERRUPT YET
004055	250040	000043*	11970	CL4:	EXCH	B,T, CU;	
004056	254520	003766*	11980		JRST	12, AAPRR	
004057	336000	005471*	12000		SKIPN	SS99	NOT MUCH POINT IF DRUM IS BUSY
004060	336000	002107*	12010		DMBY;		SET INTERPRETER RETURN FLAG
004061	350000	002111*	12020		AOS		
004062	254000	004055*	12030		JRST	CL1	
			12040				

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.

PAGE 43

CONSTANTS

004130	201004	020100	12560	BLANKS:	ASCII	?	?
004131	201004	020100	12570	ASCII	?	?	
004132	000000	000000	12580	ZERO:	0		
004133	064321	506432	12590	CR\$:	OCT	064321506432;	FIVE CARRIAGE RETURNS
004134	064240	000000	12600	CRLF:	OCT	064240000000;	CR, LF, AND ZEROS
			12610	BELF:	ASCII	?\`	
004135	034240	700000	12620				
			12630				
004136	440700	000000	12640	C3:	POINT	7 INITIAL	
004137	440700	000000	12650	C4:	POINT	7, JONNO	
004140	220600	004015•	12660	C5:	POINT	6, SWITCH, D17;	POINTER TO DISPLAY LINE NUMBER

```

12680 ; TRY TO SAVE THE SYSTEM FROM HARDWARE ERRORS
12690 ;
12700 ; NOTE THAT PROPER RECOVERY FROM MACHINE ERRORS
12710 ; DEPENDS, IN PART, ON A SPECIFIC LOADING ORDER
12720 ; FOR JOSS COMPONENTS. IN PARTICULAR, THE ORDER
12730 ; SHOULD BE: ARITH,DISTRIB, DISC, SU, CPU, DDT, IU.
12740 CHECK: CONSO APR,10000; SKIP IF NO EX MEM
          JRST CHECK1
          HRRZ J,APRR;
          CAIG J,INTBEG
          HALT 27;
          CONO APR,110007;
          MOVEI A,72
          JRST 10,KILL
          PICK UP OFFENDING LOCATION
          SKIP IF CPA ILLEGAL OP

004141 700340 010000
004142 254000 004151*
004143 550440 003766*
004144 307440 003707*
004145 040000 000027
004146 700200 110007
004147 201000 000072
004150 254400 004162*
12750 ;
12760 ;
12770 ;
12780 ;
12790 ;
12800 ;
12810 ;
12820 ;
12830 ;
12840 ;
12850 ;
12860 ;
12870 ;
12880 ;
12890 ;
12900 ;
12910 ;
12920 ;
12930 ;
12940 ;
12950 ;
12960 ;
12970 ;
12980 ;
12990 ;
13000 ;
13010 ;
13020 ;
13030 ;
13040 ;
13050 ;
13060 ;
13070 ;
13080 ;
13090 ;
13100 ;
13110 ;
13120 ;
13130 ;
13140 ;
13150 ;
13160 ;
13170 ;
13180 ;
13190 ;
13200 ;
13210 ;
13220 ;

004151 550440 003766*
004152 700340 020000
004153 254000 004176*
004154 550440 003766*
004155 307440 004144*
004156 040000 000031
004157 700200 120007
004160 200000 000074
004161 254400 004162*
12900 KILL: MOVE PP,[XWD -N,PP1,PP1-1]; RESTORE MONITOR PUSH REGIS
          MOVE S,CUI
          MOVE B,S,DU
          LDB B,CHK1
          JUMPE SETZN
          SKIPN S,DQ
          JRST CHK1
          AOS DKBY;
          SUBI S,S,Q
          CHS COM,S
          OGKM QUIT
          DISCARD A BAD USER BLOCK

004162 200740 010542*
004163 200700 000021*
004164 135040 000267*
004165 322040 004174*
004166 402000 002110*
004167 336700 002131*
004170 254000 004174*
004171 350000 002110*
004172 275700 000141*
004173 020000 000007
004174 260740 004211*
004175 254000 004446*
12950 KILL: MOVE PP,[XWD -N,PP1,PP1-1]; RESTORE MONITOR PUSH REGIS
          MOVE S,CUI
          MOVE B,S,DU
          LDB B,CHK1
          JUMPE SETZN
          SKIPN S,DQ
          JRST CHK1
          AOS DKBY;
          SUBI S,S,Q
          CHS COM,S
          OGKM QUIT
          DISCARD A BAD USER BLOCK

004176 700740 100000
004177 040000 000002
12960 ;
12970 ;
12980 ;
12990 ;
13000 ;
13010 ;
13020 ;
13030 ;
13040 ;
13050 ;
13060 ;
13070 ;
13080 ;
13090 ;
13100 ;
13110 ;
13120 ;
13130 ;
13140 ;
13150 ;
13160 ;
13170 ;
13180 ;
13190 ;
13200 ;
13210 ;
13220 ;

004200 700600 100000
004201 332000 002125*
004202 254000 004206*
004203 350000 002167*
004204 256000 004162*
004205 254400 007264*
13000 HANDLE PARITY ERRORS
13010
13020
13030
13040
13050
13060
13070
13080
13090
13100
13110
13120
13130
13140
13150
13160
13170
13180
13190
13200
13210
13220

004206 201000 000076
004207 202456 002572*
004210 254400 004162*
13000
13010
13020
13030
13040
13050
13060
13070
13080
13090
13100
13110
13120
13130
13140
13150
13160
13170
13180
13190
13200
13210
13220

004200 700600 100000
004201 332000 002125*
004202 254000 004206*
004203 350000 002167*
004204 256000 004162*
004205 254400 007264*
13000
13010
13020
13030
13040
13050
13060
13070
13080
13090
13100
13110
13120
13130
13140
13150
13160
13170
13180
13190
13200
13210
13220

004206 201000 000076
004207 202456 002572*
004210 254400 004162*
13000
13010
13020
13030
13040
13050
13060
13070
13080
13090
13100
13110
13120
13130
13140
13150
13160
13170
13180
13190
13200
13210
13220

```

13240 ; ON ENTRY REGISTER A CONTAINS THE GRONK CODE
13250 ; A GRONK IS REPORTED AT THE CONSOLE TTY BY A LINE OF DATA
13260 ; GIVING THE GRONK CODE (ONE OF THOSE LISTED BELOW), THE
13270 ; NUMBER OF THE CONSOLE GRONKED, AND TWO DATA WORDS WITH
13280 ; CONTENTS AS LISTED BELOW.
13290 ;
13300 ; 71- CKSUM FAILURE: 1) CORRECT, 2) INCORRECT
13310 ; IF THE DP OR DR HARDWARE DETECTS AN ERROR THE
13320 ; CONTENTS OF THE LEFT AND RIGHT HALVES OF
13330 ; THE INCORRECT CHECKSUM WORD CONTAIN THE RESULTS
13340 ; OF CONI INSTRUCTIONS ON THE DP AND DR.
13350 ; FAILURE TO COMPARE FINAL DATA1 REGISTER CONTENTS
13360 ; WITH COMPUTED VALUES ALSO TAKE THIS EXIT. IN THIS
13370 ; CASE THE LEFT HALF WILL BE ZERO AND THE RIGHT
13380 ; HALF WILL SHOW NO ERROR FLAG.
13390 ; 72- NOEX MEM FROM INTERPRETER
13400 ; 73- INITIALS CHECK: 1) BAD INITIALS, 2) GOOD
13410 ; 74- CPA ILL OP = PROTECT VIOLATION ON USER BLOCK REFERENCE
13420 ; 75- ILLEGAL UQO FROM INTERPRETER CODE
13430 ; 76- PARITY ERROR; 2) OFFENDING LOCATION
13440 ;
004211 200700 000021* OGKM: MOVE S.CUI
004212 200440 000000* 13450 MOVE J,A
004213 260740 006274* 13460 TSX OFFNO;
004214 260740 010343* 13470 TSX GETBUF
004215 040000 000026* 13480 HALT 26; IF SO WE ARE IN TERRIBLE TROUBLE
004216 200100 010543* 13490 MOVE C,[POINT 7.1 (E)]
13510 XMT 3,BELF
OPDEF X [3B12]
004217 000140 004135* X BELF
004220 303440 000100* 13520 CALLE J,100;
004221 200451 000000* 13530 MOVE J,0(J);
004222 200000 000011* 13540 MOVE A,J
004223 260740 004274* 13550 TSX OCTW
13560 XMT 2,BLANKS
OPDEF X [2B12]
004224 000100 004130* X BLANKS
004225 200000 000021* 13570 MOVE A,CUI
004226 260740 006700* 13580 TSX OCT
13590 XMT 10,GKM
OPDEF X [10B12]
004227 000400 004247* X GKM
13600 XMT 2,BLANKS
OPDEF X [2B12]
004230 000100 004130* X BLANKS

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
OUTPUT A GRONK MESSAGE

PAGE 46

004231	200700	000021*	13620	MOVE	S,CUI
004232	200016	002642*	13630	MOVE	A,SUM(S)
004233	260740	004274*	13640	TSX	OCTW
			13650	XMT	2,BLANKS
004234	000100	004130*	OPDEF X[2B12]		
004235	200016	002572*	13660	MOVE	A,MINT(S) :
004236	260740	004274*	13670	TSX	OCTW
			13680	XMT	3,BELF
004237	000140	004135*	OPDEF X[3B12]		
			X BELF		
			13690	XMT	3,CRLF
004240	000140	004134*	OPDEF X[3B12]		
004241	571700	777777	X CRLF		
004242	260740	010305*	13700	HRREI	S,-1
004243	200700	000021*	13710	TSX	PUTB
004244	402016	002572*	13720	MOVE	S,CUI
004245	350000	002224*	13730	SETZM	MINT(S)
004246	263740	000000	13740	AOS	CT2; ;
			13750	DONE	
004247	036172	247634	13760		COUNT ERROR
004250	455020	700000	13770	GKM:	
				ASCII	?NGRONK!\\?;
					8 CHARACTERS

```

13790 ; TABLE FOR CONVERSION TO LOGARITHMIC RANGE
004251 000000 072460
004252 000000 027340
004253 000000 013560
004254 000000 005670
004255 000000 002260
004256 000000 001130
004257 000000 000454
004260 000000 000170
004261 000000 000074
          CVTLOG: DEC 500*60
          DEC 200*60
          DEC 100*60
          DEC 50*60
          DEC 20*60
          DEC 10*60
          DEC 5*60
          DEC 2*60
          DEC 1*60
          CVTS=-.CVTLOG

13900 ; CONVERT TO LOG RANGE
13910
13920
13930
13940
13950
13960
13970
13980
13990
14000
14010
14020
14030
14040
14050
14060
14070
14080
14090
14100
14110
14120
14130
14140
14150
14160
14170
14180
          CVTL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl
          CVTLL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          TRAP TIME COPY OF CONVERT ROUTINE
          CVTLL1: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          CONVERT A WORD TO OCTAL ASCII
          CLOBBERS D,B, INPUT IS A.
          OCTW: MOVEI D,D12
          MOVE B,A
          SETZM A
          LSHC A,3
          ORI A,60
          IDPB A,C
          SJG D,OCTW1
          DONE

004262 000000 000000
004263 201100 000011
004264 311042 004250*
004265 367100 004264*
004266 254120 004262*
          CVTL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl
          CVTLL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          TRAP TIME COPY OF CONVERT ROUTINE
          CVTLL1: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          CONVERT A WORD TO OCTAL ASCII
          CLOBBERS D,B, INPUT IS A.
          OCTW: MOVEI D,D12
          MOVE B,A
          SETZM A
          LSHC A,3
          ORI A,60
          IDPB A,C
          SJG D,OCTW1
          DONE

004267 000000 000000
004270 201100 000011
004271 311042 004250*
004272 367100 004271*
004273 254120 004267*
          CVTL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl
          CVTLL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          TRAP TIME COPY OF CONVERT ROUTINE
          CVTLL1: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          CONVERT A WORD TO OCTAL ASCII
          CLOBBERS D,B, INPUT IS A.
          OCTW: MOVEI D,D12
          MOVE B,A
          SETZM A
          LSHC A,3
          ORI A,60
          IDPB A,C
          SJG D,OCTW1
          DONE

004274 201140 000014
004275 200040 000000
004276 402000 000000
          CVTL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl
          CVTLL: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          TRAP TIME COPY OF CONVERT ROUTINE
          CVTLL1: Z MOVEI C,CVTS
          CAML B,CVTLOG-1 (C)
          SJG C,-1
          JRST 2,acvtl1
          CONVERT A WORD TO OCTAL ASCII
          CLOBBERS D,B, INPUT IS A.
          OCTW: MOVEI D,D12
          MOVE B,A
          SETZM A
          LSHC A,3
          ORI A,60
          IDPB A,C
          SJG D,OCTW1
          DONE

```

	14200	;	DECODE INTERPRETER SIGNALS		
004304	250700	000021*	14210	MONENT: EXCH S,CUL; MOVE C,T-CU D,S,TM	RESTORE THE USER INDEX
004305	200100	000043*	14230	LDB C,D; SUB C,USTAT(S);	TICKS SINCE LAST ENTRY ACCUMULATE IN USER STAT BLOCK
004306	135140	000262*	14240	ADDM C,COMTIM; ADDM C,CT13;	UPDATE USERS COMPUTE TIME UPDATE COMPUTE TIME THIS MINUTE
004307	274100	000003	14250	ADDM C,CT44;	ACCUMULATE TIME COMPUTING
004310	272116	002522*	14260	LDB A,S,BLOCK;	NUMBER OF CORE BLOCKS
004311	272100	000000	14270	A,C IMUL	INTEGRATE TIME X CORE SIZE="CHARGE UNITS"
004312	272100	002165*	14280	ADDM A,S,BLOCK;	
004313	272100	002213*	14290	SETZB A,T-CU	RESET POSSIBLE IN SIGNAL
004314	135000	000264*	14300	DPB A,S,INR;	
004315	220000	000002	14310	ADD C,D	
004316	272000	000000	14320	DPB A,S,TM	
004317	403000	000043*	14330	PP,PPSAV;	RESTORE MONITOR PP REGISTER
004320	137000	000266*	14340	EXCH B,L-ETSW;	CHECK FOR WITHIN LIMIT.
004321	270100	000003	14350	CALL 5;	BAD SIGNAL FROM JOE
004322	137000	000262*	14360	JRST @ENTSW(B) ;	INTERPRET ENTRY.
004323	250740	000020*	14370		
004324	301040	000021	14380		
004325	040000	000005	14390		
004326	254021	004327*	14400		
			14410		
			14420	;	INTERPRETER SIGNAL TYPES
			14430		
004327	254000	004372*	14440	ENTSW: JRST SU;	0 - SWITCH TO USER
004330	254000	004430*	14450	JRST BUFFER;	1 - GET A BUFFER
004331	254000	004444*	14460	JRST REBUF;	2 - RETURN A BUFFER
004332	254000	004366*	14470	JRST TL;	3 - TRANSMIT BUFFER TO USER
004333	254000	004370*	14480	JRST TLSU;	4 - TRANSMIT AND SWITCH TO USER
004334	254000	004355*	14490	JRST CONT;	5 - BREAK POINT RESPONSE
004335	254000	004604*	14500	JRST DREQ;	6 - REQUEST DISC
004336	254000	004616*	14510	JRST DCONT;	7 - CONTINUE DISC ACTION
004337	254000	004625*	14520	JRST DCOMP;	8 - DISC ACTION COMPLETE
004340	254000	004760*	14530	JRST MORCOR;	9 - GET ANOTHER BLOCK OF CORE
004341	254000	005013*	14540	JRST MORCI;	10 - IMPERATIVE MORE CORE
004342	254000	004501*	14550	JRST PAGE;	11 - GET UP AND SEND PAGE HEADING
004343	254000	004446*	14560	JRST QUIT;	12 - GET USER OFF
004344	254000	004726*	14570	JRST RCOR;	13 - RETURN BLOCKS OF CORE
004345	254000	004416*	14580	JRST PAUSE;	14 - PAUSE FOR SIGNALS AND (E) SECONDS
004346	254000	004420*	14590	JRST WAIT;	15 - PAUSE (E) SECONDS LENGTH OF THE SWITCH
004347	000000	000000	14600	L,ETSW=-ENTSW+1; Z;	SPARE
			14620		
			14630	;	RETURN TO INTERPRETER
			14640		
004350	201300	002150*	14650	MONEXIT:MOVEI G,INTENT;	INTERPRETER CONTEXT
004351	250700	000021*	14660	MONE1: EXCH S,CUI	
004352	250740	000020*	14670	PP,PPSAV;	
004353	402000	002111*	14680	COMEBACK SETZM	
004354	254026	000000	14690	JRST @O(G); GO TO PROPER INTERPRETER	

004355	315100	000326*	14710	*	COMPUTATION BREAK	HAS HE HIT MAX?
004356	254000	004361*	14720	CONT:	CAME	NO
004357	020000	000007	14730		JRST	PUT CURRENT USER ON END OF LIST
004360	254000	007264*	14740		CHS	
			14750		COM.S;	
			14760		SIGPR1	
004361	350000	002113*	14770		AOS	SET TOP OF QUEUE SWITCH FOR CHS
004362	137100	000262*	14780		DPB	C,S,TM
004363	020000	000007	14790		CHS	COM.S;
004364	402000	002113*	14800		SETZM	RESET SWITCH
004365	254000	007264*	14810		JRST	
			14820		SIGPR	
			14830	*		
004366	260740	010305*	14840	*	TRANSMIT LINE TO USER	
004367	254000	004350*	14850		PUTB;	ATTACH BUFFER TO USER
004370	260740	010305*	14860	TL:	JRST	GO BACK TO INTERPRETER
004371	254000	004404*	14870		MONEXIT;	
			14880	TLSU:	PUTB;	ATTACH BUFFER TO USER
			14890		SU1	
			14900			
			14910	*		
			14920	;	SWITCH CONSOLE TO USER	
004372	3111700	004063*	14930	SU:	CAML	S,CONSOL;
004373	254000	004401*	14940		JRST	SU,5
004374	260740	010343*	14950		GETBUF	
004375	254000	004414*	14960		JRST	
004376	515040	574000	14970		HRLZI	B,574000;
004377	202044	000003	14980		MOVEM	B,3 (E) ;
004400	260740	010305*	14990		TSX	PUTB;
004401	550216	000331*	15000	SU.5:	HRRZ	E,S,BUF (S) ;
004402	402000	000000	15010		SETZM	RISIG;
004403	322200	004406*	15020		JUMPE	E,SU2;
004404	020000	000016	15030	SU1:	CHS	DSU,S;
004405	254000	007264*	15040		JRST	SIGPR1;
			15050			
004406	260740	010343*	15060	SU2:	TSX	GETBUF
004407	254000	004414*	15070		JRST	SU3
004410	504200	000004	15080		HRL	E,E
004411	202216	000331*	15090		MOVEM	E,S,BUF (S) :
004412	260740	006560*	15100		TSX	LINK BUFFER TO USER
004413	254000	007264*	15110		JRST	SWITCH GREEN
004414	020000	000017	15120		SIGPR1	
004415	254000	007264*	15130	SU3:	CHS	TO WAIT BUFFER FOR GREEN QUEUE.
			15140		JRST	SIGPR
			15150			

004416 201040 000001 *
004417 137040 000271 *

004420 020000 000020
004421 260740 006421 *
004422 270040 000004
004423 303040 250600
004424 275040 250600
004425 332000 000004
004426 506056 000141 *
004427 254000 007264 *

15170 ; WAIT FOR SIGNALS AND TIME
15180 PAUSE: MOVEI B,1
15190 DPB B,S,SIG
15200
15210
15220 ; DELAY USER FOR (E) SECONDS
15230 WAIT: CHS QP,S; TO PAUSE STATE
15240 TSX ISEC
15250 ADD B,E
15260 CALLF B,D24*D3600
15270 SUBI B,D24*D3600
15280 NO TIME SPECIFIED
15290 SKIPE E;
15300 HRLM B,S,Q(S); SAVE WAKEUP TIME IN QUEUE WORD
15310 JRST SIGPR

004430 200200 000323* 15330 ; GET A BUFFER
15340
004431 550056 000331* 15350 BUFFER: MOVE E,N,CK;
15360 B,S,BUF(S);
GET THE CHOKE NUMBER
GET TOP BUFFER LOCATION
004432 322040 004437* 15370 JUMPE: HRRZ B,BU2;
15370 B,JMP; OK - GET A BUFFER FOR HIM
GET NEXT ON LIST
004433 550041 000000 15380 HRRZ B,0(B);
15390 SCJG E,-2;
367200 004432* 15400 CHS CK,S;
004435 020000 000013 15400 JRST SIGPR1
004436 254000 007264* 15410
15420
004437 260740 010343* 15430 BU2: TSX GETBUF
15440 JRST *4;
004440 254000 004442* 15440 NO BUFFER
15450 MONEXIT
004441 254000 004350* 15450 Jrst MONEXIT
15460 CHS UC,S;
004442 020000 000005 15460 BUF1: SIGPR1
004443 254000 007264* 15470 JRST
15480
15490 ; RETURN A BUFFER
15500
004444 260740 010401* 15510 REBUF: TSX MBA1;
15520 JRST MONEXIT;
PUT BUFFER ON AVAILABLE
GO BACK TO INTERPRETER

```

15540          *           FINAL "OFF" FOR USER
15550          *           C,S,BLOCK
15560          QUIT:    LDB      D,I,POINT 6,CT48A]
15570          MOVE     ILDB     B,D
15580          ILDG     SOJG     C,-1
15590          DECREMENT BLOCK SIZE DISPLAY
15600          SUBI     B,1
15610          DPB      B,D;
15620          SETZM   MINT(S)
15630          LDB      B,S,COR
15640          JUMPE   B,Q2;
15650          SUBI     B,BBLOCK;
15660          LDB      C,S,BLOCK;
15670          SETZM   CORE(B);
15680          AOS      B
15690          SOJG     C,-2;
15700          SOS      CT39;
15710          LDB      A,S,STA
15720          SETZM   S,S(S);
15730          DPB      A,S,STA;
15740          MOVEI   A,1
15750          DPB      A,S,BLOCK;
15760          CHS      OF,S
15770          FSW      D
MOVE D,SWITCH
004473 200140 004140*  D,OFFS;
004474 606140 100000*  TRNN   D,OFFS;
004475 264000 000000*  JSR    C31
004476 200140 010545*  MOVE   D,ASCII ?OFF-?; DISPLAY OFFS ON CONSOLE TTY
004477 260740 006664*  TSX    DOF
004500 254000 007264*  JRST   SIGPR1
15830
15820
15810
15800
15790
15780
15770
15760
15750
15740
15730
15720
15710
15700
15690
15680
15670
15660
15650
15640
15630
15620
15610
15600
15590
15580
15570
15560
15550
15540

```

```

15850 ; SEND PAGE HEADING
15860 ; OUTPUT PAGE HEADING LINE TO USER. ON INPUT REGISTER E
15870 ;
15880 ; CONTAINS A BUFFER ADDRESS
15890 PAGE: MOVE C,[POINT 7,3(E)]; POINTER TO BUFFER BEGINNING
004501 200100 010546* 15910 ; PAGED ALREADY!
004502 331000 0000044 15920 ; FORM FEED
004503 010000 000014* 15930 ; CONVERT HOURS WITH LEADING BLANKS
004504 200000 000031* 15940 ; INSERT A ":";
004505 260740 006410* 15950 ; CONVERT MINUTES WITH LEADING ZEROS
004506 010000 000072* 15960 ; MOVE A,MIN
004507 200000 000032* 15970 ; CTD;
004510 260740 006403* 15980 ; XMT 2,BLANKS
004511 000100 004130* OPDEF X [2B12]
X BLANKS 15990 ; XMT 12,DATE
OPDEF X [12B12] X DATE
004512 000500 000023* 16000 ; INS "#
004513 010000 000043 16010 ; MOVE A,S
004514 200000 000016 16020 ; TSX BTA;
004515 260740 004715* 16030 ; XMT 2,BLANKS
004516 000100 004130* OPDEF X [2B12]
X BLANKS 16040 ; XMT 5,INITIALS
OPDEF X [5B12] X INITIALS
004517 000240 004136* 16050 ; XMT 5,JOBNO
OPDEF X [5B12] X JOBNO
004520 000240 004137* 16060 ; CAML S,CONSOL;
004521 311700 004372* 16070 ; XMT 4,BLANKS;
OPDEF X [4B12] SKIP IF TTY CONSOLE
X BLANKS 16080 ; CENTER PAGE NUMBER FOR JOSS CONSOLE
AOS B,PAGNO;
004522 000200 004130* 16090 ; CAIGE B,D100;
004523 350040 000000 16100 ; XMT 1,BLANKS
004524 305040 000144 ; FETCH AND INCREMENT PAGE NUMBER
; CENTER PAGE NUMBER A LITTLE
OPDEF X [1B12]
X BLANKS 16110 ; REMEMBER REG. A IS ZERO
004526 010000 000133 16120 ; CONVERT TO ASCII BASE TEN
004527 *200000 004523* 16130 ; INS "1"
004530 260740 004715* 16140 ; XMT 2,BLANKS
004531 010000 000135 16150 ;
OPDEF X [2B12]
X BLANKS 16160 ; MAKE UP HEADING MESSAGE
004532 000100 004130* 16170 ; TRANSMIT THE LINE
004533 260740 004535* ; TSX HDM;
004534 254000 004366* ; JRST TL;

```

69

-56-

```

004535 200140 004473*   16190   HDM:   FSW D;   FETCH DATA SWITCHES
004536 602140 000040   16200   MOVE D,SWITCH   TRNE
004537 254000 004551*   16210   JRST   D,PRMES;
004540 606140 200000   16220   TRNN   D,SDS;
004541 254000 004544*   16230   JRST   PG3
004542 001600 004571*   16240   XMT 34,HSDM; PUT SHUTDOWN MESSAGE IN HEADER
004543 254000 004557*   X HSDM   JRST   PG1
004544 606140 040000   16250   PG3:   TRNN   D,DCLOB
004545 254000 004550*   16260   JRST   PG2
004546 001400 004577*   X DCLOBM   XMT 30,DCLOBM;   SEND DISC CLOBBED MESSAGE
004547 254000 004557*   16290   JRST   PG1
004550 602140 000001   16300   PG2:   TRNE   D,DAM;
004551 336040 004562*   16310   PG4:   SKIPN   B,L,OPM;
004552 254000 004557*   16320   JRST   PG1;
004553 200140 010536*   16330   MOVE   D,[POINT 7,OPMSG]
004554 134000 000003   16340   ILDB   A,D
004555 136000 000002   16350   IDPB   A,C;
004556 367040 004554*   16360   SOJG   B,-2
004557 000300 004133*   X CRS   XMT 6,CRS; SIX CRS
004560 010000 000000   16380   INS   0
004561 263740 000000   16390   DONE
004562 000000 000000   16400   L,OPM:   DEC   /
004563 201004 020100   16410   OPMSC:   ASCII   /
004564 201004 020100   16420   ;   * CHARACTERS IN OPERATOR MESSAGE
004565 201004 020100   16430   ;   /; ONLY 30 CHAR
004566 201004 020100   16440   ;   SHUTDOWN MESSAGE: "PREPARE FOR JOSS SHUTDOWN."
004567 201004 020100   16450   HSDM:   BYTE   (7) 120,162,145,160,141,162,145,40,146,157
004570 201004 020100   16460   BYTE   (7) 162,7,40,112,117,123,40,163,150
004571 503454 570302   16470   BYTE   (7) 165,164,144,157,167,156,56,7
004572 713124 063336   16480   ;   DISC CLOBBED MESSAGE: "FILES ARE NOT USEABLE."
004573 710164 045236   16490   BYTE   (7) 7,106,151,154,145,163,40,141,162,145
004574 516464 071720   16500   DCLOBM:   BYTE
004575 727514 467756   16510   BYTE   (7) 40,7,156,157,164,40,165,163,7,141
004576 671340 700000   16520   BYTE   (7) 142,154,145,56
004600 715014 171312   16530   BYTE
004601 200175 667750   16540   BYTE
004602 203536 303702   16550   BYTE
004603 613314 527000   16560   BYTE

```

				DISC ACTION ROUTINES
16550	*	DREQ:	MOVEI SETZN DPR SKI PN DKBY; JRST DQS; CHS JRST AOS SPARE4; SETOM JRST	A,1 SPARE4; A,S.DU; DKBY; DREQ1 DQS; SIGPR1 DKBY; TELL JOE HE GOT THE DISC RETURN TO INTERPRETER
004604	201000 000001	16560		RESET DISC AVAILABLE FLAG
004605	402000 000000	16570		FLAG AS DISC USER
004606	137000 000267*	16580		SKIP IF DISC IS BUSY
004607	336000 002110*	16590		
004610	254000 004613*	16610		DISC BUSY - PUT ON QUEUE
004611	020000 000014	16620		SET DISC BUSY SWITCH
004612	254000 007264*	16630		TELL JOE HE GOT THE DISC
004613	350000 002110	16640		RETURN TO INTERPRETER
004614	476000 004605*	16650		
004615	254000 004350*	16660		
		16670		
004616	700740 000100	16680		
004617	254000 004622*	16690		
004620	020000 000011	16700		PI, CHDC
004621	254000 007264*	16710		DCONT1;
004622	020000 000012	16720		DCT.S;
004623	264000 003773*	16730		SIGPR1
004624	254000 007264*	16740		DIP.S;
		16750		DISC.C;
004625	403040 002110*	16760		SIGPR1
004626	137340 000267*	16770		
004627	250700 002112*	16780		INDICATE DISC IDLE
004628	336700 002131*	16790		RESET DISC FLAG
004630	254000 004635*	16800		
004631	350000 002110*	16810		SKIP IF SOMEONE IN THE QUEUE
004632	350000 000141*	16820		NONE IN QUEUE
004633	275700 000141*	16830		BUSY THE DISC
004634	020000 000007	16840		
004635	250700 002112*	16850		ACTIVATE USER
004636	135000 000265*	16860		
004637	322000 004350*	16870		COM.S;
004640	260740 006274*	16880		S,MISC
004641	254000 007264*	16890		LDB
		16900		A,S.OFR;
		16910		GET OFF FLAG
		16920		JUMPE
		16930		A,MONEEXIT;
		16940		OFENO;
		16950		SIGPR1
		16960		
		16970		
		16980		
				DISC RESTART ROUTINE
004642	700700 000100			
004643	263740 000000			
004644	275700 000141*			
004645	020000 000012			
004646	264000 004623*			
004647	263740 000000			
				NO RESTART IF DC IS BUSY
				S,S.Q
				DIP.S
				DISC.C;
				CONTINUE DISC ACTION

17000		ACCOUNT FOR DISC USAGE AT DISCARD TIME
17010	,	USES REGISTERS B,E,S; ENTERED VIA PUSHJ PP,ADIS.
17020	ADIS:	MOVEI C, ¹ MOVS B,DISC.D(C) CMN B,DISC.D(C); SKIP IF DATES ARE DIFFERENT
17030		DONE; TSX NO ACCOUNTING
17040		GETBUF IN THIS CASE THEY GO FREE
17050		DONE; MOVE B,FILE MOVEM B, ¹ D7(E)
17060		MOVE B,KEY
17070		MOVE B, ¹ D8(E)
17080		MOVE B,MINT(S)
17090		MOVE B, ¹ D9(E)
17100		MOVE B,[XWD,D9, ¹ D9]
17110		MOVE B, ¹ (E)
17120		MOVEI B,2(E); TO ADDRESS
17130		MOVE B,DISC.D; ADDRESS OF ACCOUNTING DATA
17140		MOVE S,B
17150		MOVE BLT B,4(S); MOVE TO BUFFER
17160		HRZ B,PROG
17170		HRM B,2(E)
17180		MOVE A, ¹ 6(E)
17190		MOVE TLNE A, ¹ 777777; CHECK FOR BINARY
17200		TSX TLT; NO - CONVERT FROM ASCII
17210		TLNE A, ¹ 777777
17220		MOVE F, ¹ 6(E)
17230		MOVE S,[DEC -2]; CONTEXT FOR TAPE
17240		PUTB TSX DONE
17250		
17260		
17270		
17280		
17290		
17300		
17310		
17320	TLT:	SETZM F
17330		MOVE C,[POINT 7,A]
17340	TLT1:	ILDB D,C
		CAIN D, ¹ ""; WE ARE DONE ON THE FIRST BLANK
17350		DONE
17360		ANDI D, ¹ 7
17370		IMULI F, ¹ D10
17380		ADD F,D
17390		JRST TLT1
17400		

			THE NUMBER IN A IS CONVERTED AND STUFFED IN THE STRING AT C, B IS CLOBBERED
	17420	;	
	17430	;	
004715	402000 000001	BTA:	SETZM
004716	231000 000012		IDIVI
004717	506057 000000		HRLM
004720	322000 004722*		JUMPE
004721	260740 004715*		TSX
004722	554017 000000	BTA 1:	HLRZ
004723	435000 000060		ORI
004724	136000 000002		IDPB
004725	263740 000000		DONE

004715 402000 000001 ;
004716 231000 000012 ;
004717 506057 000000 ;
004720 322000 004722* ;
004721 260740 004715* ;
004722 554017 000000 ;
004723 435000 000060 ;
004724 136000 000002 ;
004725 263740 000000 ;

BTA:
SETZM
IDIVI
HRLM
JUMPE
TSX

B
A,*D10
B,0 (PP) ;
A,BTA1;
BTA;

A,*0 (PP) ;
A,60;
A,C;

PUT REMAINDER AWAY ON PDL
END, GO TO OUTPUT
RECURSE
RECOVER DIGIT
PUT IN THE ASCII BITS
DEPOSIT IN STRING

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
RETURN BLOCKS OF CORE

PAGE 58

004726 135100 000264* 17560 RCOR: LDB C,S.BLOCK
004727 200140 010544* 17570 MOVE D,[POINT 6, CT48A]
004730 134000 000003 17580 ILDB A,D
004731 367100 004730* 17590 SCJG C,-1
004732 275000 000001 17600 SUBI A,1;
004733 137000 000003 17610 DECR A,D
004734 135100 000264* 17620 LDB C,S.BLOCK
004735 274100 000004 17630 SUB C,E
004736 305100 000001 17640 HALT C,1
004737 040000 000023 17650 ATTEMPT TO RETURN ALL USER CORE
004740 137100 000264* 17660 DBB C,S.BLOCK; SET NEW USER SIZE
004741 200140 010544* 17670 MOVE D,[POINT 6, CT48A]
004742 134000 000003 17680 ILDB A,D
004743 367100 004742* 17690 SCJG C,-1
004744 271000 000001 17700 ADDI A,1
004745 137000 000003 17710 INCREMENT SIZE DISTRIBUTION
004746 135100 000264* 17720 LDB C,S.BLOCK
004747 135000 000263* 17730 LDB A,S.COR
004750 275000 000020 17740 SUBI A,BBLOCK
004751 270100 000000 17750 ADD C,A
004752 270100 000004 17760 ADD C,E;
004753 275100 000001 17770 SUBI C,1
004754 402002 000272* 17780 SETZM CORE(C);
004755 367200 004753* 17790 SCJG E,-2
004756 260740 005110* 17800 TSX SRR;
004757 254000 004350* 17810 MONEXIT JRST

004760 201040 000000 17830 MORCOR: MOVEI B,0; PRESET TO DENY REQUEST
 004761 135100 000264* 17840 LDB C,S.BLOCK; CURRENT # OF BLOCKS
 004762 311100 000312* 17850 CAML C,N.CB; COMPARE WITH MAX ALLOWABLE
 004763 254000 004350* 17860 JRST MONEEXIT; ENOUGH - DENY REQUEST
 004764 200140 010544* 17870 MOVE D,[POINT 6,CT48A]
 004765 134040 000003 17880 ILDB B,D COUNT DOWN TO PROPER BYTE
 004766 367100 004765* 17890 SOJG C,-1;
 004767 275040 000001 17900 SUBI B,1;
 004770 137040 000003 17910 DPB B,D;
 004771 134040 000003 17920 ILDB B,D ONE LESS AT THIS SIZE
 004772 271040 000001 17930 ADDI B,1 ONE MORE AT NEXT SIZE
 004773 137040 000003 17940 DPB B,D;
 17950 IS BLOCK JUST ABOVE HIM FREE?
 17960
 004774 135100 000264* 17980 LDB C,S.BLOCK; CURRENT # OF BLOCKS
 004775 135140 000263* 17990 LDB D,S.COR; CURRENT BLOCK LOCATION
 004776 275140 000020 18000 SUBI D,BB BLOCK
 004777 270140 000002 18010 ADD D,C;
 005000 350000 000002 18020 AOS C; INDEX TO NEXT HIGHER BLOCK
 005001 311140 000313* 18030 CAML D,N.C;
 005002 254000 005020* 18040 JRST MOR6;
 005003 332043 000272* 18050 SKIPE B,CORE(D);
 005004 254000 005020* 18060 JRST MOR6;
 005005 661040 400000 18070 TLO B,400000;
 005006 202043 000272* 18080 MOVEM B,CORE(D);
 005007 137100 000264* 18090 DPB C,S.BLOCK;
 18100
 005010 201040 000001 18110 MOVEI B,1; NOW HE HAS ONE MORE
 005011 260740 005110* 18120 TSX SRR TELL JOE HE GOT ONE
 005012 254000 004350* 18130 JRST MONEEXIT; TELL INTERPRETER HE GOT ONE
 18140
 18150 ; IMPERATIVE REQUEST FOR MORE CORE
 18160
 005013 135100 000264* 18170 MORCI: LDB C,S.BLOCK SKIP IF TOO BIG FOR MACHINE
 005014 315100 000313* 18180 CAMGE C,N.C; SKIP IF DRUM IS BUSY
 005015 254000 004764* 18190 JRST MOR7
 005016 201040 000000 18200 MOVEI B,0
 005017 254000 004350* 18210 JRST MONEEXIT; DENY REQUEST
 18220
 18230 NOW WE MUST SET UP A SWAP
 005020 137100 000264* 18250 MOR6: DPB C,S.BLOCK; SET NEW SIZE
 005021 336000 002107* 18260 SKIPN DMBY; SKIP IF DRUM IS BUSY
 005022 254000 005025* 18270 JRST QDM,S;
 005023 020000 000021 18280 CHS SIGPR
 005024 254000 007264* 18290 JRST
 18300
 005025 275100 000001 18310 MOR7: SUBI C,1; SIZE TO SHIP OUT
 005026 476000 003331* 18320 SETOM DMIN;
 005027 260740 003234* 18330 OSWAP;
 005030 020000 000006 18340 CHS QC,S;
 005031 254000 007264* 18350 JRST SIGPR

18380 ; FIND A FREE BLOCK OF CORE
18390 ; (OF REQUIRED SIZE ANYWHERE)
18400 ;
18410 ; C- REQUIRED # OF BLOCKS, E IS DESTROYED
18420 ; B= INDEX OF TOP FOUND BLOCK, D= CORE CELL (=0) OF LAST BLOCK
18430 ; ROUTINE SKIPS ON NO SUCCESS
18440 ;
005032 201040 000000 FINDB: MOVEI B,0
005033 201200 000000 MOR2: MOVEI E,0
005034 332141 000272, MOR2.5: SKIPE D,CORE(B)
005035 254000 005046, JRSI MOR3.5
005036 350000 000004 AOS E
005037 317100 000004 CAMG C,E;
005040 263740 000000 DONE;
005041 350000 000001 AOS B
005042 315040 000313, CAMGE B,N,C;
005043 254000 005034, JRSI MOR2.5
005044 350017 000000 AOS 0 (PP);
005045 263740 000000 DONE
005046 201200 000000 MOR3.5: MOVEI E,0
005047 254000 005041, JRSI MOR3
18590
18600 ZZ=.

```
18620 ; FIND A FREE BLOCK OF THE REQUIRED SIZE  
18630 ; (C = REQUIRED SIZE)  
18640  
18650 MOR1: TSX FINDB  
18660 JRST M2; FOUND GO TO MOVE  
18670  
18680 ; FIND A FREE BLOCK BELOW USER  
18690  
18700 ; EXITS WITH B= INDEX TO FREE BLOCK  
18710  
18720 MOR4: LDB B,S,COR  
18730 SUBI B,BBLOCK  
18740 JUMPE B,MOR5;  
18750 SUBI B,1  
18760 MOVE D,CORE(B)  
18770 TLINE D,200000;  
18780 JRST MOR5;  
18790 SKIPN D  
18800 JRST M4;  
18810 SJGE B,MOR41;  
18820
```

005064	135040	000263*	18840	18850	,	FIND A FREE BLOCK ABOVE USER; C= # OF REQUIRED BLOCKS
005065	275040	000020	18860	M0R5:	LDB	B,S,COR
005066	270040	000002	18870		SUBI	B,BBLOCK
005067	275040	000002	18880		ADD	B,C;
			18890		SUBI	B,2
005070	311040	000313*	18900	M0R51:	CAML	B,N,C
005071	254000	005020*	18910		JRST	M0R6;
005072	200141	000272*	18920		MOVE	D,CORE (B)
005073	603140	200000	18930		TLINE	D,200000
005074	254000	005020*	18940		JRST	M0R6;
005075	336000	000003	18950		SKIPN	D
005076	254000	005135*	18960	JRST	M5;	FOUND ONE: GO TO MOVE
005077	344040	005070*	18970	A0JA	B,M0R51;	GO TO LOOK AT NEXT BLOCK
			18980			

19000 ; MOVE USER TO FREE AREA
 19010 ;
 19020 ; C= # OF BLOCKS IN NEW USER AREA
 19030 ; B= CORE MAP INDEX OF LAST BLOCK IN "TO" REGION
 19040 ;
 005100 661140 4000000 M2: TLO D,4000000; BUSY THE HIGH BLOCK
 005101 202141 000272* MOVEM D,CORE(B);
 005102 137100 000264* DPB C,S,BLOCK;
 005103 275100 000001 19080 SUBI C,1;
 005104 274040 000002 19090 SUB B,C;
 005105 135200 000263* 19100 LDB E,S,COR
 005106 275200 000020 19110 SUBI E,BBLOCK;
 005107 200300 000004 19120 MOVE G,E;
 005110 260740 005073* 19130 TSX BLTC;
 005111 402006 000272* 19140 SETZM CORE(G);
 005112 350000 000006 19150 AOS G
 005113 367100 005111* SOJG C,M25
 005114 260740 005050* 19160 TSX RUL;
 005115 254000 005010* 19170 JRST M1;
 19180 ; RESET USER LOCATIONS
 19190 ; SUCCESS RETURN TO JOE
 19200 ; MOVE DOWN MULTIPLE USERS (B=FREE BLOCK INDEX)
 19210 ;
 19220 ; C= NEW SIZE
 19230 ;
 005116 135240 000263* 19240 M4: LDB F,S,COR
 005117 275240 000020 19250 SUBI F,BBLOCK
 005120 137100 000264* 19260 DPB C,S,BLOCK;
 005121 270100 000005 19270 ADD C,F
 005122 200000 000002 19280 MOVE A,C;
 005123 275100 000002 19290 SUBI C,2
 005124 274100 000001 19300 SUB C,B;
 005125 200200 000001 19310 MOVE E,B;
 005126 271200 000001 19320 ADDI E,1;
 005127 260740 005073* 19330 TSX BLTC;
 005130 260740 005050* 19340 TSX RUL;
 005131 201040 000001 19350 MOVEI B,1
 005132 250000 000001 19360 EXCH A,B
 005133 137000 000314* 19370 DBA,S,IU;
 005134 254000 005010* 19380 JRST M1;

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
PROVIDE MORE CORE FOR USERS BLOCK

PAGE 64

005135	137100	000264*	19410	*	MOVE USER (S) UP
005136	135200	000263*	19420	*	B= FREE BLOCK INDEX, C= NEW SIZE
005137	275200	000020	19430	*	
005140	270200	000002	19440	M5:	DPB C,S,BLOCK; SET NEW SIZE
005141	275200	000001	19450		LDB E,S,COR
005142	261740	000001	19460		SUBI E,BBLOCK
005143	200001	000271*	19470		ADD E,C
005144	202001	000272*	19480		SUBI E,1
005145	370000	000001	19490		PUSH PP,B;
005146	313040	000004	19500		MOVE A,CORE-1 (B)
005147	19510	005143*	19510	M51:	MOVEM A,CORE (B);
005150	201100	000001	19520		SOS B
005151	137100	000314*	19530		SAVE FREE BLOCK LOCATION
005152	262740	000001	19540		MOVE THE CORE MAP
005153	271040	000020	19550		CAMLE B,E
005154	240040	000012	19560		JRST M51
005155	271200	000020	19570		MOVEI C,1
005156	240200	000012	19580		DPB C,S,IU;
005157	370000	000001	19590		POP PP,B
005160	200141	000000	19600		ADDI B,BBLOCK
005161	202141	002000	19610		ASH B,↑D10
005162	313040	000004	19620		ADDI E,BBLOCK
005163	254000	005157*	19630	M52:	ASH E,↑D10;
005164	260740	005050*	19640		LAST WORD TO BE MOVED
005165	254000	005010*	19650		SOS B
			19660		MOVE D,0 (B)
			19670		MOVEM D,2000 (B)
			19680		CAMLE B,E
			19690		JRST M52
			19700		TSX RUL;
			19710		JRST M1;
				RELOC ZZ	RESET USER LOCATIONS RETURN TO JOE

19730 ; RESET USER LOCATIONS
19740 , SCAN CORE MAP AND UPDATE USER LOCATIONS
19750 ,
19760 RUL:
19770 EXCH S,MISC
19780 MOVEI B,O
19790 SKIPN E,CORE(B); SKIP IF BUSY BLOCK
RUL1: AOA B,RUL2; BUMP TO NEXT
19800 TLINE E,200000
AOJA R,RUL2; DONT MESS WITH DRUM CORE
19810 LDB S,S,UR; USER FOR THIS BLOCK
19820 MOVE E,B
19830 ADDI E,BBLOCK
19840 DPB E,S,COR
19850 LDB E,S,BLOCK
19860 F,S,STA
19870 LDB F,QDM,S
19880 CAIN E,1
19890 SUBI B,E;
19900 ADD CRGE B,N,C
19910 RUL1; AROUND FOR NEXT
19920 JRST S,MISC
19930 EXCH DONE
19940
19950
19960

19980	:	BLOCK TRANSFER CORE		
19990	,	MOVE BLOCKS DOWN - USERS AND CORE MAP		
20000	,	C= BLOCKS TO BE MOVED		
20010	,	B= "TO" BLOCK INDEX		
20020	,	E= "FROM" BLOCK INDEX		
20030	,	CLOBBERS F,E,B.		
20040	,			
20050	,			
005073	504040 000004	BLTC:	HRL B,E;	BLT POINTER WORD
005074	200200 000001	20060	MOVE E,B	
005075	270200 000002	20070	ADD E,C	
005076	275200 000001	20080	SUBI E,1	
005077	200240 000001	20090	MOVE F,R;	F MAY BE DESTROYED BY INTERRUPT
005100	270240 010551*	20110	ADD F,[XWD CORE,CORE]	
005101	251244 000272*	20120	BLT F,CORE (E);	MOVE CORE MAP
005102	270040 010552*	20130	ADD B,[XWD BBLOCK,BBLOCK]	
005103	240040 000012	20140	ASH B, ¹ D10	
005104	271200 000020	20150	ADDI E,BBLOCK	
005105	240200 000012	20160	ASH E, ¹ D10	
005106	251044 001777	20170	BLT B,2000~1 (E)	
005107	263740 000000	20180	DONE	
20190		20200		SET RELOCATION REGISTER
20210		20220	SRR:	LDB D,S,COR
20230		20240	ASH D, ¹ D10	
20250		20260	ADD D,[OCT 100000700000];	
20270		20280	LDB I,S,BLOCK	
20290		20300	SUBI I, ¹ 1	
20310		20310	ASH I, ¹ D10	
005110	135140 000263*	005111	MOVSS I	
005111	240140 000012	005112	ADD D,I	
005112	270140 010553*	005113	DATAO APR,D;	SET RELOCATION
005113	135400 000264*	005114	DONE	
005114	275400 000001	005115		
005115	240400 000012	005116		
005116	207000 000010	005117		
005117	270140 000010	005120		
005120	700140 000003	005121		
263740 000000				

```

20330          COMPACT CORE
              20340          COMPACT CORE
              20350          COMPACT:SETZB
                                B,H
                                SKIPN CORE (B)
                                ACS H;
                                COUNT FREE BLOCKS IN H
              20360          20370          ACS
              20380          20390          ACS B
                                CAMGE B,N,C
                                JRST ^4
              20400          20410          TSX COMP1;
                                DO A SINGLE COMPACT
              20420          20430          DONE
              20440          20450          JRST -2
              20460          ;      DO ONE COMPACT
              20470          COMP1: T$X FM;
              20480          DONE; FIND A MOVE
              20490          MOVE NONE
              20500          ADD D,C
              20510          TSX BEGINNING OF FREED AREA
              20520          SETZM DO THE MOVE
              20530          CORE(D); ZERO FREED CORE
              20540          AOS
              20550          CAMGE D,G
              20560          JRST COMP2
              20570          MOVE E,C
              20580          ADDM C,CT50
              20590          ADDM E,T7+5
              20600          TSX RESET USER LOCATIONS
              20610          RUL; 0 (PP)
              20620          AOS TAKE "DID ONE" EXIT
              20630          DONE;

```

```

20650    : FIND A CORE MOVE
20660    :
20670    : EXITS +1 IF NONE, +2 IF A MOVE IS FOUND
20680    : ON ENTRY H= # OF FREE BLOCKS
20690    : ON EXIT:
20700    :     B= 1ST FREE LOCATION ("TO" BLOCK)
20710    :     E= 1ST NON-ZERO BLOCK ("FROM")
20720    :     C= # OF BLOCKS TO MOVE
20730    :     G= # OF FIRST UNAFFECTED BLOCK
20740    :     F AND I ARE CLOBBERED
20750    :
20760    :
20770    : MOVEI   B,0
20780    : MOVE   D,CORE (B)
20790    : JUMPE  D,FM2;
20800    : AOS    B
20810    : CAME   B,N.C;
20820    : JRST   FM1
20830    : DONE;
20840    :
20850    : MOVEI   B,0
20860    : MOVE   D,CORE (B)
20870    : SKIPE  E,B;
20880    : JRST   FM1
20890    : AOS    I
20900    : SONG   H
20910    : DONE;
20920    : CAME   E,N.C;
20930    : JRST   FM3
20940    : DONE;
20950    :
20960    : MOVEI   C,E;
20970    : MOVE   F,CORE (C)
20980    : JUMPE  F,FM7
20990    : TLNE   F,2000000;
21000    : JRST   FM7;
21010    : AOS    C
21020    : CAME   C,N.C;
21030    : JRST   FM5
21040    :
21050    : MOVEI   G,C;
21060    : SUB    C,E;
21070    : SKIPE  C
21080    : DONE;
21090    : AOS    0 (PP);
21100    : DONE;
21110    :

005152   201040 000000
005153   200141 000272
005154   322140 005161
005155   350000 000001
005156   312040 000313
005157   254000 005153
005160   263740 000000
005161   200200 000001
005162   200400 000007
005163   332004 000272
005164   254000 005173
005165   350000 00004
005166   377000 000010
005167   263740 000000
005170   312200 000313
005171   254000 005163
005172   263740 000000
005173   200100 000004
005174   200242 000272
005175   322240 005203
005176   603240 200000
005177   254000 005203
005200   350000 000002
005201   312100 000313
005202   254000 005174
005203   200300 000002
005204   274100 000004
005205   337000 000002
005206   263740 000000
005207   350017 000000
005210   263740 000000

```

ON ENTRY H= # OF FREE BLOCKS

ON EXIT:

- B= 1ST FREE LOCATION ("TO" BLOCK)
- E= 1ST NON-ZERO BLOCK ("FROM")
- C= # OF BLOCKS TO MOVE
- G= # OF FIRST UNAFFECTED BLOCK
- F AND I ARE CLOBBERED
- EXITS +1 IF NONE, +2 IF A MOVE IS FOUND
- SEARCH FOR 1ST UNUSED BLOCK
- SEARCH FOR 1ST UNUSED BLOCK
- END TEST
- EXIT: NO FREE BLOCKS IN CORE
- B IS "TO" LOCATION (1ST FREE BLOCK)
- SEARCH FOR 1ST NON-ZERO BLOCK
- END TEST
- WE HAVE ALL FREE BLOCKS TOGETHER
- END TEST
- CORE IS ALREADY COMPACT
- E IS "FROM" LOCATION (1ST NON-ZERO)
- MOVEI B,0
- MOVE D,CORE (B)
- JUMPE D,FM2;
- AOS B
- CAME B,N.C;
- JRST FM1
- DONE;
- MOVEI B,0
- MOVE D,CORE (B)
- SKIPE E,B;
- JRST FM1
- AOS I
- SONG H
- DONE;
- CAME E,N.C;
- JRST FM3
- DONE;
- MOVEI C,E;
- MOVE F,CORE (C)
- JUMPE F,FM7
- TLNE F,2000000;
- JRST FM7;
- AOS C
- CAME C,N.C;
- JRST FM5
- DONE;
- MOVEI G,C;
- SUB C,E;
- SKIPE C
- DONE;
- AOS 0 (PP);
- DONE;
- MOVEI G,C;
- SAVE # OF FIRST UNAFFECTED BLOCK
- C HAS # OF BLOCKS TO MOVE
- NO NEED TO MOVE ZERO BLOCKS
- TO SECOND EXIT--MOVE FOUND

21130 * FIND A USER TO BRING IN
 21140 SELSWP: SKIPN ON
 21150 SKIPE N.DRM
 21160 JRST .+2;
 21170 DONE SELECT FOR SWAP IF "ON" USERS OR SOME ON DRUM
 21180 MOVE C,[POINT 5,SI1]; TABLE OF CANDIDATES
 21190 LLDB B,C
 21200 CAIN B,END,S
 21210 DONE NONE REQUIRED
 21220 MOVEI A,1
 21230 HRZR B,S,QUE(B); QUEUE HEADER
 21240 JUMP B,SS1; JUMP IF LIST EMPTY
 21250 HRZR S,B
 21260 SUBI S,S,Q;
 21270 LDB D,S,COR;
 21280 LDB F,S,GR
 21290 JUMPN F,SS3;
 21300 LDB F,S,STA;
 21310 CAIN F,ON,S
 21320 JRST SS10;
 21330 JUMPE D,SS4;
 21340 HRZR B,0(B);
 21350 JUMPE B,SS1;
 21360 AOUA A,SS2
 21370
 21380 * USER * IN S, STATE IN F, PLACE IN QUEUE IN A
 21390
 21400 FIND A FREE BLOCK FOR HIM
 21410 *
 21420 SET "IN" CANDIDATE
 21430 HRRZM S,DMIN;
 21440 HRLM A,DMIN;
 21450 LDB C,S,BLOCK;
 21460 TSX FINDB;
 21470 JRST SS15;
 21480 TRY TO FIND SCATTERED FREE BLOCKS
 21490
 21500 SETZB B,E
 21510 SETIPN CORE(B)
 21520 AOS E;
 21530 COUNT FREE BLOCKS
 21540 AOS B
 21550 CAMGE B,N,C
 21560 JRST SS6
 21570 CAMGE E,C;
 21580 JRST SS17;
 21590 TSX COMPACT;
 21600 LDB C,S,BLOCK
 21610 TSX FINDB
 21620 JRST SS15;
 21630 JRST SS17;
 005245 403040 000004
 005246 336001 000272*
 005247 350000 000004
 005250 350000 000001
 005251 315040 000313*
 005252 254000 005246*
 005253 315200 000002
 005254 254000 005305*
 005255 260740 005122*
 005256 135100 000264*
 005257 260740 005032*
 005260 254000 005277*
 005261 254000 005305*

005262	200040	000313*	21650	*	DEAL WITH "ON" SEPARATELY
005263	275040	000001	21660	SS10:	MOVE B,N,C
005264	336001	000272*	21670	SUBI B,1	
			21680	SKIPN CORE (B)	
005265	263740	000000	21690	DONE;	
005266	365040	005264*	21700	SOJGE B,-2;	
005267	200040	010555*	21710	MOVE B,[POINT 5,S12];	
005270	200340	010556*	21720	MOVE H,[TSX SS(8)];	
005271	260740	005472*	21730	TEST ROUTINE	
005272	263740	000000	21740	FIND	
005273	562700	003331*	21750	CANT FIND ONE BEST GET COMPUTING	
005274	135100	000264*	21760	SET DMIN;	
005275	260740	003234*	21770	SET NO IN REQUIRED	
005276	263740	000000	21780	LDR C,S,BLOCK	
			21790	OSWAP	
			21800	DONE	
005277	135100	000264*	21810	*	READY FOR "IN" SWAP; B= TOP OF IN BLOCK
005300	275100	000001	21820	SS15:	LDB C,S,BLOCK
005301	274040	000002	21830	SUBI C,1	
005302	202040	003324*	21840	SUB B,C	
005303	260740	003333*	21850	MOVEM B,DMIBK;	
005304	263740	000000	21860	TSX ISWAP	
			21870	DONE	
			21880		
			21890		
			21900	*	FIND A LOW PRIORITY USER OF THE PROPER SIZE
005305	200240	000002	21910	SS17:	MOVE F,C
005306	200040	010557*	21920	MOVE B,[POINT 5,S11]	
005307	200340	010556*	21930	MOVE H,[TSX SS(8)]	
005310	260740	005472*	21940	MOVE TSX FIND	
005311	254000	005327*	21950	JRST SS30;	
005312	254000	005274*	21960	JRST SS11;	
			21970	FOUND - GO TO START OUT	
005313	135300	000263*	21980	LDB G,S,COR	
005314	336000	000006	21990	SKIPN G;	
005315	263740	000000	22000	DONE	
005316	315140	000005	22010	CAME D,F;	
005317	263740	000000	22020	DONE	
005320	350017	000000	22030	AOS 0 (PP);	
005321	263740	000000	22040	DONE	
			22050		
			22060		

```

22080 ; FIND CANDIDATE (S) FOR DRUM
22090 22100 S10: BYTE (5) OP..S,ABG..S,DSU..S,GR..S,DQ..S,CK..S,DIP..S,DCT..S,END..S
005322 407563 261324
005323 237400 000000
005324 407563 374000
005325 407563 261324
005326 223770 000000

22110 S11: BYTE (5) OP..S,ABG..S,DSU..S,GR..S,DQ..S,CK..S,DIP..S,DCT..S,COM..S,END..S
22120 S12:BYTE (5) OP..S,ARG..S,DSU..S,GR..S,DQ..S,CK..S,DIP..S,DCT..S,COM..S,END..S

22130
22140 , FIND MULTIPLE USERS TO KICK OUT
22150 , E= # OF FREE BLOCKS, F= REQUIRED SIZE TO FREE
22160
22170 SS30: MOVE B,[POINT 5,S10] H,[TSX SS35]; TEST ROUTINE
22180 MOVE G,DMNR
22190 MOVE SUB F,E;
22200 SUB FIND
22210 TSX JRST SS40; REQUIRED BLOCKS TO FREE
22220 FIND NO SUCCESS - TRY COMPUTE QUEUE
22230
22240 , START OUT, POSSIBLY MULTIPLE, USERS
22250
22260 SS32: POP G,S; GET FIRST USER TO WRITE
22270 LDR C,S,BLOCK
22280 TSX OSWAP;
22290 MOVEM G,DMNR;
22300 TLLNE G,777777; SKIP IF ONLY ONE TO WRITE
22310 JRST SS33
22320 LDB B,S,BLOCK; BLOCKS WRITTEN
22330 HRRZ S,ADMIN; USER TO COME IN
22340 LDB C,S,BLOCK; BLOCKS TO COME IN
22350 CAMGE B,C; SKIP IF WE ARE FREEING ENOUGH SPACE
22360 AOS SS98; FIRE COMPACT ACTION
22370 DONE; DONE WITH SWAP SELECTION
22380
22390 SS33: MOVEI C,1
22400 AOS SS98; SIGNAL CORE COMPACT NEEDED
22410 SS34: POP G,S
22420 LDB D,S,BLOCK
22430 LDB B,S,COR
22440 SUBI B,BBLOCK
22450 DPB C,S, ID; BUSY ALL BLOCKS INVOLVED IN SWAP
22460 AOS B
22470 SOJG D,-2
22480 TLNE G,777777
22490 JRST SS34
22500 DONE;

005365 135000 000263
005366 336000 000000
005367 263740 000000
005370 274240 000003
005371 261300 000116
005372 333000 000005
005373 263740 000000
005374 350017 000000
005375 263740 000000

22510
22520 SS35: LDB A,S,COR
22530 SKIPN A NOT IN CORE TRY NEXT
22540 DONE; F,D
22550 SUB G,S; SAVE AS USER TO WRITE
22560 PUSH SKIPL F; SKIP IF WE HAVE SELECTED ENOUGH
22570 DONE; AOS 0 (PP) TAKE SUCCESS EXIT
22580
22590
22600
22610

```

```

22630      NOW WE MUST KICK OUT A COMPUTE USER
22640      F= REQUIRED BLOCKS TO FREE. DMIN: R=USER FOR IN,L= PLACE IN QUEUE
22650      G= CURRENT LIST TO WRITE
22660      E= # OF FREE BLOCKS
22670

005376  200700 003331:  SS40:      MOVE    S,DMIN
005377  135040 000261:  LDB     B,S,STA;
005400  201140 000001:  MOVEI   D,1
005401  201000 000001:  MOVEI   A,1
005402  306040 000007:  CAIN    B,COM,S;
005403  554000 000016:  HLRZ    A,S;
005404  200100 002124:  MOVE    C,COM;
005405  322100 005442:  JUMPE   C,SS45;
005406  201040 000037:  MOVEI   B,END,S
005407  261740 000001:  PUSH    PP,B;
005410  550700 000002:  HRRZ    S,C
005411  275700 000141:  SUBI    S,S,O
005412  261740 000016:  PUSH    PP,S;
005413  350000 000003:  AOS    D
005414  506157 000000:  HRLM    D,0,(PP);
005415  550102 000000:  HRRZ    C,0,(C)
005416  326100 005410:  JUMPN   C,SS41;
005417  262740 000016:  POP    PP,S;
005420  306700 000037:  CAIN    S,END,S
005421  254000 005442:  JRST    SS45;
005422  554140 000016:  HLRZ    D,S;
005423  317140 000000:  CAMG    D,A
005424  254000 005437:  JRST    SS44;
005425  135040 000263:  LDB    B,S,COR
005426  3222040 005417:  JUMPE   B,SS42;
005427  274240 000001:  SUB    F,B;
005430  261300 000016:  PUSH    G,S;
005431  327240 005417:  JUMPG   F,SS42;
005432  262740 000001:  POP    PP,B
005433  302040 000037:  CAIE    B,END,S
005434  254000 005432:  JRST    T7+6;
005435  350000 002446:  AOS    SS32;
005436  254000 005335:  JRST    COUNT COMPUTE USERS KICKED OUT
                           SUCCESS--GO FOR FIRST WRITE
                           23010
                           23020
                           23030
                           23040
                           23050
                           23060
                           23070
                           23080
                           23090
                           23100
                           23110
                           23120
                           23130
                           23140
                           23150
                           23160
                           23170
                           23180

                           SS41:      POP    PP,B
                           CATE   B,END,S
                           JRST    -.2
                           MOVE   B,[XWD 0,DMNR]
                           MOVEM B,DMNR;
                           DONE;

                           SS42:      MOVE   B,N,S-1
                           LDB    B,S,COR
                           JUMPE B,TE2;
                           ASH    B,D10
                           MOVE   B,2(B);
                           CAME  B,MINT(S)
                           HALT  22;
                           SOJGE S,TE1
                           DONE;

                           SS43:      RESET WRITE QUEUE HEADER
                           CANNOT SET UP WRITE--NEED TO COMPUTE
                           23100
                           TEST:  MOVEI   S,N,S-1
                           LDB    B,S,COR
                           JUMPE B,TE2;
                           ASH    B,D10
                           MOVE   B,2(B);
                           CAME  B,MINT(S)
                           NOT THE SAME AS CORE RECORD
                           23110
                           TE1:   MOVEI   S,N,S-1
                           LDB    B,S,COR
                           JUMPE B,TE2;
                           ASH    B,D10
                           MOVE   B,2(B);
                           CAME  B,MINT(S)
                           NOT THE SAME AS CORE RECORD
                           23120
                           23130
                           23140
                           23150
                           23160
                           23170
                           23180

```

23200 , COMPACT CORE FOR SWAP IN
23210 SS90: TSX TEST
23220 TSX COMPACT;
23230 TSX TEST
23240 MOVE S.DMIN
23250 SETZM SS99;
23260 LDB C.S.BLOCK
23270 FINDB
23280 TSX
23290 JRST SS15
23300 DMBY
23310 SETZM
23320 DONE;
23330 SS98: Z; SET IF MULTI WRITE
23340 SS99: Z; SET TO REQUEST COMPACTION OF CORE AND IN SWAP
005456 260740 005445*
005457 260740 005122*
005460 260740 005445*
005461 200700 003331*
005462 402000 005471*
005463 135100 000264*
005464 260740 005032*
005465 254000 005277*
005466 402000 002107*
005467 263740 000000
005470 000000 000000
005471 000000 000000

23360 ; FIND FIRST USER IN STATES LISTED AT POINTER IN B
23370 ; WHICH SATISFIES TEST IN H (THAT SKIPS ON SUCCESS).
23380 , C IS DESTROYED, S IS SET TO FOUND USER
23390 ,
23400 FIND: ILDB C,B; INDEX TO STATE HEADER
23410 CAIE C,END..S; SKIP IF END
23420 JRST F1 FAILURE EXIT
23430 DONE;
23440
23450 F1: HRRZ C,S,QUE (C); LIST HEADER
23460 JUMPE C,FIND; JUMP IF LIST EMPTY
23470 F2: HRRZ S,C
23480 SUBI S,S,Q; COMPUTE STATION INDEX
23490 LDB D,S,BLOCK
23500 DO TEST
23510 XCT H;
23520 JRST F3
23530 AOS 0 (PP)
23540 DONE; TEST SUCCEEDS
23550 F3: HRRZ C,0 (C); TEST ROUTINE FAILS; GET NEXT ON LIST
23560 JUMPN C,F2;
23570 JRST FIND; GO FOR NEXT LIST
23580
23590

005512	402000	000324*	23610	;	FORCE USERS OFF	
005513	201000	200010	23620	BOF:	SETZM	N,SON;
005514	436000	004535*	23630		MOVEI	A,SDS+DOAF;
005515	505000	000003	23640		ORM	A,SWITCH;
005516	700600	001030	23650		HRLI	A,3;
005517	200100	000115*	23660		CONO	PI,1000+CH630;
005520	201700	000047	23670		MOVE	C,SG,L;
005521	135040	000261*	23680		MOVEI	S,N,S-1;
005522	135240	000267*	23690		LDB	B,S,STA;
005523	326240	005536*	23700		LDB	F,S,DU;
005524	200240	010563*	23710		JMPN	F,BOF5;
005525	134200	000005	23720		MOVE	F,(POINT 5,BOF10)
005526	306200	000037	23730	BOF2:	ILDB	E,F;
005527	254000	005533*	23740		CAIN	E,END,S;
005530	316200	000001	23750		JRST	BOF3;
005531	254000	005536*	23760		CAMN	E,B;
005532	254000	005525*	23770		JRST	BOF5;
005533	540000	000016	23780		JRST	BOF2;
005534	202002	000045*	23790	BOF3:	HRR	A,S;
005535	350000	000002	23800		MOVEM	A,SLIGTBL(C);
005536	365700	005521*	23810		AOS	C;
005537	202100	000115*	23820	BOF5:	SOGE	S,BOF1;
005540	700600	002030	23830		MOVEM	C,SG,L;
005541	263740	000000	23840		CONO	PI,20000+CH630;
			23850		DONE	TURN ON 630 INTERRUPTS
			23860			
			23870			
			23880			
			23890			
005542	460370	000000	23900	BOF10:	BYTE	(5) OF,S,TOF,S,END,S
			23910			

				PROCESS USERS WAITING FOR BUFFERS
005543	275700	000141	23930	; 23940
005544	260740	010343	23950	MSGPR: SUBI S,S.Q; COMPUTE USER NUMBER
005545	263740	000000	23960	GETBUF
005546	311700	004521	23970	DONE;
			23980	SHOULDNT HAPPEN THE FIRST TIME
				SKIP IF TRY CONSOLE
005547	254000	005555	23990	CAML S,CONSOL;
005550	515040	574000	24000	MSGPR1 JRST MSGPR1
005551	202044	000003	24010	BRLZI B,574000; LEFT ARROW CODE
005552	260740	010305	24020	MOVEM B,3 (E); STASH IN BUFFER
005553	020000	000016	24030	PUTB TSX
005554	263740	000000	24040	DSU,S CHS
005555	504200	000004	24050	DONE HRL E,E
005556	202216	000331	24060	MSGPR1: MOVEM E,S.BUF (S); LINK TO USER
005557	260740	006560	24070	SG; SWITCH TO GREEN
005560	263740	000000	24080	DONE
			24090	
			24100	PROCESS THE PAUSE QUEUE
			24110	
005561	544116	000000	24120	PQP: HLR C,0 (S); GET TIME COUNT
005562	322100	005571	24130	JUMPE C,PQP2
005563	540716	000000	24140	HRR S,0 (S); NEXT ON LIST
005564	260740	006421	24150	ISEC TSX
005565	302040	000002	24160	CAIE B,C
005566	254000	005571	24170	PQP2; JRST TIME NOT UP YET
005567	020000	000007	24180	CHS ITS TIME - ACTIVATE HIM.
005570	553016	000141	24190	HRRZS ZERO REQUEST TIME
005571	326700	005561	24200	PQP2: JUMPN JUMP IF MORE IN PAUSE QUEUE
005572	263740	000000	24210	DONE
			24220	

```

24240 ; PROCESS DISC INTERRUPT SIGNAL
24250
24260
24270
24280
24290
24300
24310
24320
24330
24340
24350
24360
24370
24380
24390
24400
24410
24420
24430
24440 ;
24450
24460
24470
24480
24490
24500
24510
24520
24530
24540
24550

DISCP: CAIE D,3; SKIP IF DISC RE-ENTRY
        JRST DISC2
        SETZM DISC.S
        USR DISC.C
        DONE

DISC2: SKIPE DSS; SKIP IF NOT SKULKING
        JRST SKULK
        SETZM S,DIP;
        HALT 10;
        SUBI S,S.Q;
        CHS COM.S;
        CAIN D,2;
        TSX ADIS;
        SETZM DISC.S;
        TSX TRST;
        DONE

H10:   ; DISC INTERRUPT BUT NO DISC USER
        COMPUTE USER NUMBER
        ACTIVATE USER FOR INFORMATION TRANSFER
        SKIP IF NOT A DISCARD
        ACCOUNT FOR DISC USE
        RESET THE SIGNAL
        RESTART THE TAPE

DISC1: ; SEARCH FOR GRONKED USERS
        JRST GRONK:
        GR1: MOVEI S,N,S-1
              LDB B,S,GK
              JUMPN B,GR3
        GR2: SOJGE S,GR1
              SETZM CKER
              DONE

        GR3: MOVEM S,CUI
              MOVEI A,71
              JRST KILL

005573 302140 000003
005574 254000 005600*
005575 402000 000000
005576 264000 004646*
005577 263740 000000

005600 332000 006101*
005601 254000 006032*
005602 336700 002127*
005603 040000 000010
005604 275700 000141*
005605 020000 000007
005606 306140 000002
005607 260740 004650*
005610 402000 005575*
005611 260740 003532*
005612 263740 000000

```

24570 ; THESE ROUTINES ARE CALLED BY THE INTERPRETER TO SET (SSIG),
24580 ; RESET (RSIG), AND TEST (TSIG) THE INTER-CONSOLE SIGNAL BITS.
24590 ; ENTRY IS VIA PUSHJ 0, ROUTINE.
24600 ; REGISTERS LISTED BELOW ARE PRESUMED FREE AND NOT SAVED.
24610 U=1
24620 V=2
24630 W=3
24640 X=4
24650 Y=5
24660 Z=6
24670 ; GET SIGNAL TABLE INDEX
24680 ; INPUT: "TO" USER IN CUI
24690 ; "FROM" USER IN U
24700 ; RETURNS: "TO" INDEX IN Z
24710 ; "FROM" BIT IN V FOR SETTING AND TESTING
24720 ; DESTROYS: W
24730 ;
24740 ;
24750 ;
24760 ;
24770 GTI: MOVE V,U
24780 ASH V,-5; HIGH ORDER PART OF "FROM"
24790 MOVE Z,CUI
24800 IMULI Z,S,M
24810 ADD Z,V;
24820 MOVE W,U
24830 ANDI W,37;
24840 MOVEI V,1
24850 ASH V,0 (W); SHIFT TO PROPER POSITION (HIGH ON LEFT)
24860 POPJ 0,0
24870 ;
24880 ;
24890 ;
24900 ;
24910 ; THE BIT "TO" CURRENT USER "FROM" (U) IS RESET
24920 RSIG: PUSHJ GTI;
24930 MOVE W,TTT(Z)
24940 TDZ W,V
24950 MOVEM W,TTT(Z)
24960 POPJ 0,0
24970 ;
24980 ;
24990 ;
25000 ;
25010 ; THE BIT "TO" CURRENT USER "FROM" (U) IS TESTED.
25020 ; REGISTER V IS NON-ZERO IF BIT IS SET.
25030 TSIG: PUSHJ GTI
25040 MOVE W,TTT(Z)
25050 TDNN W,V
25060 SETZM V
25070 POPJ 0,0
005624 200100 000001
005625 240100 777773
005626 200300 000021.
005627 221300 000002
005628 270300 000002
005630 200140 000001
005631 200140 000001
005632 405140 000037
005633 201100 000001
005634 240103 000000
005635 263000 000000
005636 260000 005624.
005637 200146 005667.
005640 630140 000002
005641 202146 005667.
005642 263000 000000
005643 260000 005624.
005644 200146 005667.
005645 616140 000002
005646 402000 000002
005647 263000 000000

```

25100 ; SET A SIGNAL BIT
25110 ; THE BIT "TO" (U) "FROM" CURRENT USER IS SET
25120 ;
25130 ; SSIG: EXCH U,CUI
25140 PUSHJ GTI
25150 MOVE W,TTT (Z)
25160 TDO W,V
25170 MOVEM W,TTT (Z)
25180 EXCH U,CUI
25190
25200 ; RE-ENABLE SIGNALLED USER IF WAITING
25210 ;
25220 EXCH PP,PPSAV; GET MONITOR PUSH REGISTER
25230 MOVE Z,S; SAVE REGISTER S
25240 MOVE S,U; *TO* USER TO S
25250 LDB W,S,SIG
25260 SKIP W;
25270 CHS COM,S;
25280 MOVE S,Z;
25290 EXCH PP,PPSAV; RESTORE PP
25300 POPJ 0,0
25310
25320
25330
25340 ;
25350 ; TABLE OF INTER-CONSOLE SIGNALS
25360 ;
25370 ; ONE WORD PER 32 STATIONS PER STATION
25380 ; BITS ACROSS THE WORD CORRESPOND TO THE "FROM" CONSOLE
25390 ; AND WORDS OR GROUPS DOWN THE TABLE TO THE "TO" CONSOLE.
25400
25410 TTR: ZBLOK S,M*N,S
REPEAT S,M*N,S,
005667 000000 000000
Z 25420

```

				MIDNIGHT DISC SKULKER
006007	332000 006101*	25440	25450	
006010	254000 006013*	25460	25470	SKIPE DSS JRST RESK
006011	332000 002110*	25480	25490	SKIPE DKBY JRST MDSS2
006012	254000 006030*	25500	25510	START OR RESTART SKULK
006013	700700 000100	25520	25530	RESK: CONSZ PI,CHDC;
006014	254000 006030*	25540	25550	JRST MDSS2
006015	402000 006102*	25550	25560	SETZM SKT;
006016	350000 006103*	25560	25570	AOS SKR;
006017	402000 000001	25570	25580	SETZM B
006020	336000 006101*	25580	25590	SKIPN DSS
006021	505040 000001	25590	25600	HRLI B,1; AOS DSS;
006022	350000 006101*	25600	25610	AOS DKBY;
006023	350000 002110*	25610	25620	BUSY THE DISC
006024	541040 000144	25620	25630	REQUEST SKULK FROM DISC PROCESSOR
006025	202040 000000	25630	25640	MOVEM JSR
006026	264000 005576*	25640	25650	B ACTION DISC.C
006027	263740 000000	25650	25660	DONE
006030	350000 006102*	25660	25670	AOS SKT;
006031	263740 000000	25680	25690	DONE

```

25710 ; DISC ACTION COMPLETE
25720 SKULK: SETZN; SKR;
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25730 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK1;
          DISC ERROR
          SKIP IF RECORD READY
25740 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25750 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK1;
          DISC ERROR
          SKIP IF RECORD READY
25760 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25770 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK1;
          DISC ERROR
          SKIP IF RECORD READY
25780 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25790 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK1;
          DISC ERROR
          SKIP IF RECORD READY
25800 ; DISC NOT USING DC
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25810 ; CONVERT ASCII TO BINARY
          TLINE A,777777
          TSX TLT;
          TLINE A,777777
          MOVEM F,DBUF(B)
          ADDI B,5
          CAIGE B,D134
          JRST SK3
          MOVE S,(DEC -2);
          CONTEXT FOR TAPE
          MOVE E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          SKIP IF NO MORE RECORDS
          UNKNOWN ENTRY FROM DISC ROUTINES
25820 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25830 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25840 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25850 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25860 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25870 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25880 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25890 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25900 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25910 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25920 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25930 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25940 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25950 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
25960 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
25970 SK4: SETZN; DISC.S
          MOVE B,RESULT
          CAIN B,D12
          JRST SK5;
          DISC ERROR
          SKIP IF RECORD READY
25980 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
25990 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26000 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26010 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26020 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26030 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26040 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26050 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26060 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26070 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26080 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26090 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26100 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26110 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26120 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26130 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26140 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26150 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26160 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
26170 ; CONTEXT FOR TAPE
          MOVE S,(DEC -2);
          MOVEI E,DBUF
          MOVEI TSX
          START TO TAPE
          PUTB;
          GO TO RESTART THE TAPE
          DISCI;
          JRST TSX
          XMT 10,SK11; SAY SKULK COMPLETE
          OPDEF X[10R12]
          SET IF WE ARE SKULKING
          SET WHEN TAPE OUTPUT COMPLETE
          SET IF DISC USING DC FOR SKULK
006032 402000 006103*
006033 200040 000000
006034 306040 000014
006035 254000 006072*
006036 302040 000144
006037 254000 006055*
006040 201040 000011
006041 200001 000000
006042 603000 777777
006043 260740 004704*
006044 603000 777777
006045 202241 006041*
006046 271040 000005
006047 305040 000206
006050 254000 006041*
006051 200700 010550*
006052 201200 006045*
006053 260740 010305*
006054 254000 005610*
006055 302040 000310
006056 040000 000021
006057 260740 010422*
006060 000400 006077*
006061 402000 005610*
006062 402000 006101*
006063 336700 002131*
006064 254000 006070*
006065 275700 000141*
006066 020000 000007
006067 263740 000000
006070 402000 002110*
006071 263740 000000
006072 260740 010422*
006073 000400 006075*
006074 254000 006061*
006075 472364 051626
006076 526311 300000
006077 516272 546226
006100 202371 300000
006101 000000 000000
006102 000000 000000
006103 000000 000000

```

JOS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
SHUTDOWN PROCEDURE

PAGE 82

26190 * SHUTDOWN JOSS SYSTEM
26200 SDP: SETZB D,N,SON;
26210 MOVEI C,1 ALLOW NO MORE ON
26220 C,SWITCH;
26230 MOVEI C,QM-S.QUE; BE SURE A MESSAGE IS GOING OUT
26240 OR D,S.QUE-1 (C) ; TEST ALL BUT QUEUE AND OF
26250 SOJG C,--1
26260 SKIP D; SKIP IF ALL IS QUIET
26270 DONE ; STILL SOME ACTION
26280 HRREI S,-1;
26290 TSX ODIS TTY CONTEXT
26300 TSX TWAIT
26310 CONO MTC,1400; END FILE LOG
26320 TSX TWAIT
26330 CONO MTC,7400; BACK OVER IT
26340 TSX CMESS
26350 XMT 23,TERM; SYSTEM ALL DOWN
OPDEF X [23B12] .
X TERM
006123 001140 006130* CONO PI,1000+CHAPR; TURN OFF PROCESSOR INTERRUPTS
006124 700600 001001 26370 SETZM SWITCH; BE SURE BEEP WONT INTERFERE
006125 402000 006106* JSR SHUT; TURN OFF ALL CONSOLES
006126 264000 000000 26390
006127 254000 006127* 26400 JRST .
006130 516632 352212 26410
006131 465011 040646 TERM: ASCII ?SYSTEM HAS SHUTDOWN?
006132 202471 052650
006133 422372 747000

			PROCESS DISTRIBUTOR SIGNALS
26440	;	PRSIG:	MOVEI G,0; HRR2 S,SIGTBL(G); B,SIGTBL(G); S,S.OK
26450		SP1:	CALLE WE ARE IGNORING THIS GUY JRST TEST FOR WITHIN LIMIT B,1D5; BAD SIGNAL FROM IRWIN
26460		H6:	HALT GET STATE LDB F,S,STA; C,T2(B); LDB SP3; XCT T3(C); CONO PI,1000+CH630; AOS G; TURN OFF 630 INTERRUPTS CAML SG,L; BUMP ENTRY COUNT COMPARE WITH LENGTH
26470		SP3:	JRST DO IT CONO PI,2000+CH630; SP1; TURN ON 630 INTERRUPTS Jrst GO BACK FOR NEXT ITEM SETZM SG,L; ZERO THE SIGNAL LIST CONO PI,2000+CH630; HRRZ S,FAKE HLRZ B,FAKE TRZN B,400000; SKIP IF PSEUDO SIGNAL
26480		SP2:	DONE CALLE B,4; SKIP IF LEGITIMATE SIGNAL CALLE S,N,S; SKIP IF LEGITIMATE STATION
26490			DONE CALLE S,N,S; SKIP IF LEGITIMATE STATION LDB F,S,STA; C,T2(B); GET STATE LDB XCT DO THE ACTION ROUTINE SETZM FAKE; DONE FLAG DONE
26500			
26510			
26520			
26530			
26540			
26550			
26560			
26570			
26580			
26590			
26600			
26610			
26620			
26630			
26640			
26650			
26660			
26670			
26680			
26690			
26700			
26710			
26720			
26730			
26740			
26750			
26760			
26770			

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
TABLES FOR SIGNAL INTERPRETATION

PAGE 84

006173	350705	006200*	26790	T2:	POINT	7,T1(F),6;	TABLE OF POINTER TO COLUMNS IN T1
006174	260705	006200*	26800		POINT	7,T1(F),D13	
006175	170705	006200*	26810		POINT	7,T1(F),D20	
006176	100705	006200*	26820		POINT	7,T1(F),D27	
006177	010705	006200*	26830		POINT	7,T1(F),D34	
26840				RADIX	10	ROUTINE NAME TABLE FOR GIVEN SIGNAL AND STATE	
26850				;	ROUTINE ARE: TO,IN,ON,OFF,CR		
26860				;			
26870				;			
26880				;			
26890				;			
T1=.							
006200	004020	100402	26900	BYTE	(7) 1,1,1,1,1;	TOF	
006201	004020	104400	26910	BYTE	(7) 1,1,1,9,0;	ON	
006202	000140	104400	26920	BYTE	(7) 0,6,1,9,0;	RC	
006203	064020	104400	26930	BYTE	(7) 13,1,1,9,0;	RI	
006204	064020	104400	26940	BYTE	(7) 13,1,1,9,0;	RIB	
006205	064120	104400	26950	BYTE	(7) 13,5,1,9,0;	UC	
006206	064140	107400	26960	BYTE	(7) 13,6,1,15,0;	QC	
006207	064100	107400	26970	BYTE	(7) 13,4,1,15,0;	COM	
006210	000000	000000	26980	BYTE	(7) 0,0,0,0,0;	CU	
006211	064140	107000	26990	BYTE	(7) 13,6,1,14,0;	DCT	
006212	064140	107000	27000	BYTE	(7) 13,6,1,14,0;	DIP	
006213	060120	104400	27010	BYTE	(7) 12,5,1,9,0;	CK	
006214	064400	107000	27020	BYTE	(7) 13,16,1,14,0;	DQ	
006215	000020	104406	27030	BYTE	(7) 0,1,1,9,3;	GR	
006216	054020	104400	27040	BYTE	(7) 11,1,1,9,0;	DSU	
006217	064020	104400	27050	BYTE	(7) 13,1,1,9,0;	ABG	
006220	064100	107400	27060	BYTE	(7) 13,4,1,15,0;	QP	
006221	064140	107400	27070	BYTE	(7) 13,6,1,15,0;	QDM	
006222	064160	104000	27080	BYTE	(7) 13,7,1,8,0;	QM	
006223	004021	200402	27090	BYTE	(7) 1,1,10,1,1;	OF	
				RADIX	8		
27110				;			
27120				;			
27130				;			
006224	040000	000020	27140	HALT	20:	0 - MACHINE ERROR HALT	
006225	255000	000000	27150	NOP;	1 - IGNORE SIGNAL		
006226	255000	000000	27160	NOP;	2 - UNUSED		
006227	260740	006457*	27170	TSX	3 - GENERAL, CR RESPONSE		
006230	260740	006427*	27180	TSX	4 - CHANGE TO RI STATE		
006231	260740	006434*	27190	TSX	5 - CHANGE TO RIB STATE		
006232	260740	006454*	27200	TSX	6 - RECORD IN SIGNAL		
006233	260740	006441*	27210	TSX	7 - SEND QUEUE MESSAGE		
006234	260740	006261*	27220	TSX	8 - OFF FROM USER IN THE QUEUE		
006235	260740	006274*	27230	TSX	9 - OFF FROM ALL OTHERS		
006236	260740	006326*	27240	TSX	10 - ON FROM OFF STATION		
006237	260740	006537*	27250	TSX	11 - TO FROM STATION IN DSU		
006240	260740	006542*	27260	TSX	12 - TO FROM CHOKE STATION		
006241	260740	006555*	27270	TSX	13 - TO FROM ALL OTHERS		
006242	260740	006323*	27280	TSX	14 - OFF FROM A DISCING USER		
006243	260740	006321*	27290	TSX	15 - OFF FROM MORE CORE		
006244	260740	006436*	27300	TSX	16 - INTERRUPT FROM DISC QUEUE		
006245	000000	000000	27310	Z;		SPARE	

H20=T3

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
SEND MESSAGES TO ALL IN THE QUEUE PAGE 85

006246	551240	000001	27340	SAQM:	HRRZI	F,1;	INITIALIZE PLACE IN QUEUE
006247	550140	002137*	27350		HRRZ	D,0M;	ADDRESS OF FIRST
006250	550700	000003	27360	SA1:	HRRZ	S,D	
006251	275700	000141*	27370		SUBI	S,S,Q;	COMPUTE USER INDEX
006252	260740	010343*	27380		TSX	GETBUF	
006253	263740	000000	27390		DONE		
006254	260740	006342*	27400		TSX	SQM;	SEND MESSAGE
006255	550143	000000	27410		HRRZ	D,0 (D) ;	
006256	3500000	000005	27420		AOS	F;	NEXT USER
006257	326140	006250*	27430		JUMPN	D,SA1;	BUMP PLACE IN QUEUE
006260	263740	000000	27440		DONE		GO AROUND IF MORE USERS

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
OFF-SIGNAL ROUTINES

PAGE 86

```

006261 370000 002164*    27460      OFF0:   SOS      CT28;
006262 402016 002572*    27470      SETZM   MINT(S) ;
006263 020000 000023    27480      CHS      OF,S
                                                FSW H

006264 200340 006125*    MOVE H,SWITCH   TRNN   H,OFFS;
                                                JSR     C31;
                                                MOVE   H,S
                                                SKIP   QM;
                                                SAQM
                                                S,H
006265 606340 100000    27500      MOVE H,SWITCH   TRNN   DONT RE-ENABLE IF BLAST OFF
006266 264000 004475*    27510      JSR     C31;
006267 200340 000016    27520      MOVE   H,S
006270 332000 002137*    27530      SKIP   QM;
                                                SEND MESSAGES TO ALL IN QUEUE
006271 260740 006246*    27540      TSX
006272 200700 000007    27550      MOVE   S,H
006273 254000 006276*    27560      JRST   OFF
006274 370000 002160*    27570      OFFNQ: SOS      OFF FROM USER NOT IN QUEUE
006275 020000 000000    27580      USERS; CHS     TOF,S
006276 200216 000331*    27590      MOVE   E,S.BUF(S);
                                                E,OF1;
006277 322200 006302*    27600      JUMPE TSX     STATION BUFFER HEADER
006300 260740 010365*    27610      MBA;   JUMP IF NO MORE BUFFERS AVAILABLE
006301 254000 006276*    27620      OFF;   PUT BUFFER ON AVAILABLE
                                                CONTINUE TILL ALL ARE PUT AWAY
                                                TURN ON A USER FROM THE QUEUE IF PROPER
006302 336000 002137*    27630      ?
006303 263740 000000    27640      ?
                                                TURN ON A USER FROM THE QUEUE IF PROPER
                                                SKIP IF SOME IN QUEUE
006304 200200 002160*    27650      OFF1:  SKIPN  QM;
                                                DONE
006305 311200 000324*    27660      MOVE   E,USERS
006306 263740 000000    27670      CAML   E,N.SON;
                                                SKIP IF USERS < MAX
                                                DONE
006307 550700 002137*    27680      OF2:   MOVE   E,USERS
                                                CAML   E,N.SON;
                                                SKIP IF USERS < MAX
                                                DONE
006310 275700 000141*    27690      OF2:   MOVE   E,USERS
                                                CAML   E,N.SON;
                                                SKIP IF USERS < MAX
                                                DONE
006311 332016 000331*    27700      HRRZ   S,QM;
                                                SUBI   S,S,Q;
                                                S,BUF(S)
006312 263740 000000    27710      SKIP
                                                DONE;
006313 020000 000001    27720      CHS     ON,S;
                                                CT2B;
006314 370000 002164*    27730      SOS     DECREMENT # IN QUEUE
006315 350000 002160*    27740      AOS     TEST FOR NONE IN THE QUEUE
006316 332000 002137*    27750      SKIP
                                                TSX
006317 260740 006246*    27760      SAQM
                                                DONE
006320 263740 000000    27770      ?
                                                LDB     SEND MESSAGES TO ALL IN QUEUE
006321 135000 000267*    27780      OFF5:  LDB     JUMP IF NOT USING DISC
006322 322000 006274*    27790      A,S.DU
                                                A,OFFNO;
006323 201000 000001    27800      MOVEI  A,1
006324 137000 000265*    27810      DPB     FLAG USER OFF DURING DISC TRANSFER
006325 263740 000000    27820      DONE

```

ON-SIGNAL ROUTINES

006326	550200	002160*	27900	ON.R:	HRRZ	E.USERS;	GET NUMBER OF USERS
006327	315200	000324*	27910		CANGE	E,N,SON;	BIGGER THAN MAX?
006330	254000	006337*	27920		JRST	ONS;	NO, GO TO TURN ON.
006331	020000	000022	27930		CHS	QM,S;	CHANGE STATE TO "QUEUE MESSAGE"
006332	350240	002164*	27940		AOS	F,CT28;	COUNT AND FETCH QUEUE *
006333	260740	010343*	27950		TSX	GETBUF;	GET A BUFFER
006334	263740	000000	27960	DONE:		,	NONE, FORGET IT
006335	260740	006342*	27970		TSX	SQM;	SEND QUEUE MESSAGE
006336	263740	000000	27980	DONE		,	
006337	020000	000001	27990	ONS:	CHS	ON,S;	CHANGE STATE TO "ON"
006340	350000	002160*	28010		AOS	USERS;	COUNT USERS
006341	263740	000000	28020	DONE		,	

```

28040      SQM:      MOVE C,[POINT 7,MINT(S)]
006342 200100 010564* 28050      INS "Q"
006343 010000 000121 28060      INS "F"
006344 010000 000043 28070      MOVE CTDDB;
006345 200000 000005 28080      TSX
006346 260740 006405* 28090      MOVE QUEUE * IN FOR INITIALS
006347 010000 000040 28100      INS "
006350 305240 000012 28110      CAIGE F,D10
006351 010000 000040 28120      INS "
006352 200000 000005 28130      MOVE A,F; RECOVER QUEUE #
006353 200100 010565* 28140      MOVE C,[POINT 7,QM2,6]
006354 260740 006410* 28150      TSX CONVERT AND DEPOSIT QUEUE #
006355 200100 010566* 28160      MOVE C,[POINT 7,QM4]
006356 200000 000031* 28170      MOVE A,HR
006357 260740 006410* 28180      TSX CONVERT AND DEPOSIT HOURS
006360 133000 000002 28190      CTDDB; SKIP PAST THE COLON
006361 200000 000032* 28200      IBP
006362 260740 006403* 28210      MOVE A,MIN
006363 030000 006371* 28220      TSX
                                         CTD; SEND
                                         QM1; "YOU ARE # IN THE QUEUE"
                                         FSW A
006364 200000 006264*      MOVE A,SWITCH
006365 602000 200000 28240      TRNE A,SDS;
006366 030000 010512* 28250      SEND SDM;
006367 260740 010305* 28260      TSX PUTB;
006370 263740 000000 28270      DONE
                                         IF SHUTTING DOWN
                                         SEND SHUT DOWN MESSAGE TO QUEUE
                                         SEND TO USER
                                         QM3=.

006371 006372*000011 28290      XWD *+1,QM3--1
006372 064241 203416 28300      OCT 64241203416;
006373 442207 246632 28310      OCT /HH:MM/
006374 202635 772500 28320      OCT 202635772500;
006375 607454 520334 28330      OCT 607454520334;
006376 727334 262744 28340      OCT 727334262744;
006377 202613 020322 28350      OCT 202613020322;
006400 671016 464312 28360      OCT 671016464312;
006401 203436 5622752 28370      OCT 2034365622752;
006402 625341 500000 28380      OCT 625341500000;
                                         QUEU
                                         E."CR"
                                         QM3=.
```

28420 ; (A) ARE CONVERTED TO CHARACTERS AT POINTER IN C, B DESTROYED
006403 231000 000012 28430 CTD: IDIVI A,[↑]D10;
006404 254000 006414; CTDI; OUTPUT HIGH ORDER ZEROS
006405 231000 000012 28440 CTDDB: IDIVI A,[↑]D10;
006406 322000 006416; JUMPE A,CTD2;
006407 254000 006414; CTDI1 CONVERT TWO DIGITS TO ASCII
006410 231000 000012 284500 CTDDB: IDIVI A,[↑]D10;
006411 326000 006414; JUMPN A,[↑]CTD1
006412 201000 000040 28520 MOVEI A," ";
006413 254000 006415; JRST ^{↑2}
006414 435000 000060 28540 CTD1: ORI A,60
006415 136000 000002 28550 IDPB A,C
006416 435040 000060 28560 ORI B,60
006417 136040 000002 28580 IDPB B,C
006420 263740 000000 28590 DONE
28600 , COMPUTE BINARY SECONDS FROM CLOCK CELLS
28610 ,
28620 ISEC: MOVE B,HR
006421 200040 000031* 28630 IMULI B,[↑]D60
006422 221040 000074 28640 ADD B,MIN
006423 270040 000032* 28650 IMULI B,[↑]D60
006424 221040 000074 28660 ADD B,SEC
006425 270040 000033* 28670 DONE
006426 263740 000000 28680

100

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
IN-SIGNAL ROUTINES PAGE 90

006427	135040	000267*	28700	IN10:	LDB	B,S.DU	
006430	332000	000001	28710		SKIPE	B;	SKIP IF NOT A DISC USER
006431	254000	006454*	28720		JRST	IN50;	FROM DISC USER: FLAG IT
006432	020000	000003	28730	IN11:	CHS	RI,S;	CHANGE TO REQUEST IN STATE
006433	254000	006452*	28740		JRST	IN99	
006434	020000	000004	28750				
006435	254000	006452*	28760	IN20:	CHS	RIB,S	
			28770		JRST	IN99	
006436	402000	000001	28780	IN30:	SETZM	B	
006437	137040	000267*	28800		DPB	B,S.DU;	RESET DISC FLAG
006440	254000	006432*	28810		JRST	IN11	
006441	260740	010343*	28820				
006442	254000	006452*	28830	IN40:	TSX	GETBUF	
006443	201240	000001	28840		JRST	IN99;	FORGET IT IF NO BUFFER
006444	550040	002137*	28850		MOVEI	F,1;	INITIALIZE QUEUE COUNT
006445	306056	000141*	28860		HRZR	B,QM	
006446	254000	006451*	28870		CAIN	B,S.Q (S);	SKIP IF NOT THE ONE OF INTEREST
006447	550041	000009	28880		JRST	*3	
006450	344240	006445*	28890		HRZR	B,0 (B)	
006451	260740	006342*	28900		AQJA	F,*-3;	COUNT PLACE IN QUEUE
006452	350000	002170*	28910		TSX	SQM;	SEND HIM THE MESSAGE
006453	263740	000000	28920	IN99:	AOS	CT27;	COUNT INS WHICH RESULT IN ACTION
			28930		DONE		
006454	201000	000001	28940	IN50:	MOVEI	A,*1	
006455	137000	000266*	28950		DPB	A,S.INR;	FLAG INTERRUPT SIGNAL
006456	254000	006452*	28960		JRST	IN99	
			28970				
			28980				

006457	350000 002173*	29000	CR.R:	AOS	CT31;	COUNT CARRIAGE RETURNS
006460	350000 002442*	29010		AOS	T7+2	
006461	550216 000331*	29020		HRZ	E,S,BUF (S) :	GET BUFFER LOCATION
006462	574144 000002	29030		HLRE	D,2 (E) :	GET CHARACTER COUNT
006463	335000 000003	29040		SKIPGE	D	
006464	462000 000003	29050		SETCM	D:	COMPLEMENT IF TTY STATION
006465	272140 002174*	29060		ADDM	D,CT25;	GROSS SUM
006466	260740 006507*	29070		TSX	CVTLIN;	FORM DISTRIBUTION
006467	350001 002332*	29080		AOS	CPIL (B)	
006470	550056 002522*	29090		HRRZ	B,USTAT (S) :	TICKS SINCE LAST CR
006471	221040 000074	29100		IMULI	B,*D60;	SCALE
006472	264000 004262*	29110		JSR	CVTL	
006473	350002 002320*	29120		AOS	DCPI (C) :	COUNT COMP/INTERACTION IN RANGE
006474	260740 006421*	29130		TSX	I,SEC;	CURRENT TIME IN SECS
006475	554116 002522*	29140		HLRZ	C,USTAT (S) :	LAST GREEN TIME
006476	516056 002522*	29150		HRLZM	B,USTAT (S) :	THIS GREEN TIME
006477	274040 000002	29160		SUB	B,C	
006500	335000 000001	29170		SKIPGE	B	
006501	271040 250600	29180		ADDI	B,*D24 * D3600;	MIDNIGHT CORRECTION UNITS ARE NOW 1/10TH MINUTE
006502	221040 000012	29190		IMULI	B,*D10;	
006503	264000 004262*	29200		JSR	CVTL	
006504	350002 002300*	29210		AOS	DTIM (C) :	INTERACTION TIME DISTRIBUTION
006505	020000 000002	29220		CHS	RC,S;	CHANGE STATE TO CR
006506	263740 000000	29230		DONE		
		29240				
		29250	:			CONVERT TO LINEAR RANGE
		29260				
006507	201040 000000	29270		CVTLIN:	MOVEI	B,0
006510	201100 000005	29280			MOVEI	C,5
006511	317140 000002	29290		CVT1:	CMG	D,C
006512	263740 000000	29300		DONE		
006513	271100 000005	29310		ADDI	C,5	
006514	305040 000011	29320		CAIGE	B,*D9	
006515	344040 006511*	29330		AQJA	B,CVT1	
006516	263740 000000	29340		DONE		
		29350				

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
TRANSMISSION-OVER-SIGNAL ROUTINES

PAGE 92

006517	336016 000331*	29370	TO:	SKIPN	HALT IF NO BUFFER
006520	040000 000007	29380	H7:	7;	NO SUCH BUFFER
006521	350000 002171*	29390		AOS	COUNT LINES
006522	350000 002443*	29400		AOS	T7+3
006523	550216 000331*	29410		HRRZ	E,S,BUF(S);
006524	550144 000002	29420		HRRZ	CHARACTER COUNT
006525	272140 002175*	29430		D,2(E)	D,CT26;
006526	260740 006507*	29440		ADDM	FIND DISTRIBUTIONAL RANGE
006527	350001 002344*	29450		TSX	CVTLIN;
006530	204056 000331*	29460		AOS	CPOL(B)
006531	430056 000331*	29470		MOVS	PICK UP BUFFER HEADER
006532	322040 006536*	29480		B,S,BUF(S);	SEE IF LEFT=RIGHT
006533	260740 010365*	29490		XOR	JUMP IF LAST BUFFER
006534	260740 000000	29500		B,TO1;	PUT BUFFER ON AVAILABLE LIST
006535	350017 000000	29510		TSX	INITIATE TRANSMISSION
006536	263740 000000	29520		CALL	BUMP TO SECOND EXIT
			TO1:	DONE	
006537	260740 006517*	29530		TO:	TO PREAMBLE
006540	260740 006560*	29540		TSX	LAST BUFFER SO CHANGE TO GREEN
006541	263740 000000	29550		SG;	
		29560		DONE	
006542	260740 006517*	29570		TO:	"TO" PREAMBLE
006543	254000 006552*	29580		TSX	ALWAYS UNCHOKE ON LAST BUFFER
006544	200200 000322*	29590		JR9T	GET UNCHOKE *
006545	550056 000331*	29600		MOVE	
006546	322040 006553*	29610		HRRZ	
006547	550041 000000	29620		JUMPE	
006550	363200 006546*	29630		B,T03;	
006551	263740 000000	29640		HRRZ	
		29650		B,0(B)	
		29660		SOJLE	
				E,-2	
				DONE;	
006552	260740 010365*	29670		TO:	STAY CHOKED (# BUFFERS > CK #)
006553	020000 000005	29680		TSX	
006554	263740 000000	29690		CHS	
				DONE	
006555	260740 006517*	29700		TO99:	CHANGE STATE TO UNCHOKE
006556	260740 010365*	29710		TO:	
006557	263740 000000	29720		TSX	
		29730		MBA;	
				DONE	

006560	550056	000331*	29750	SG:	HRRZ	B,S,BUF (\$);	POINTER TO BUFFER
006561	200200	004130*	29770	MOVE	E,BLANKS		
006562	202201	000003	29780	MOVEM	E,3 (B);	BLANK FIRST TEXT WORD	
006563	200200	000001	29790	MOVE	E,B		
006564	271200	000022	29800	ADDI	E,BUFSIZ;	LAST BUFFER LOCATION	
006565	271040	000003	29810	ADDI	B,3;	BUMP DOWN TO FIRST TEXT WORD	
006566	504040	000001	29820	HRL	B,B		
006567	350000	000001	29830	AOS	B;	MAKE UP BLT CONTROL WORD	
006570	251044	000000	29840	BLT	B,0 (E);	BLANK THE BUFFER	
006571	260740	006421*	29850	TSX	ISEC		
006572	554116	002522*	29860	HLRZ	C,USTAT (\$)		
006573	274040	000002	29870	SUB	B,C;	TASK TIME: CR TO SG	
006574	335000	000001	29880	SKIPGE	B		
006575	271040	250600	29890	ADDI	B,D24*D3600	DISPLAY UNITS ARE 1 SECOND	
006576	221040	000074	29900	IMULI	B,D60;		
006577	264000	004262*	29910	JSR	CVTL		
006600	350002	002452*	29920	AOS	T8 (C)		
006601	020000	000015	29930	CHS	GR,S;	SWITCH TO GREEN STATE	
006602	260740	000000	29940	CALL	C28;	SWITCH TO USER	
006603	263740	000000	29950		DONE		

```

006604 402000 000000    29970   QCTR:      SERZM    A
006605 322040 006611*    29980   JUMPE     B,--4
006606 550041 000000    29990   HRZRZ    B,0 (B)
006607 350000 000000    30000   AOS
006610 326040 006606*    30010   JUMPN    B,--2
006611 263740 000000    30020   DONE
006612 550040 002132*   HRRZ B,GR
006613 260740 006604*   TSX QCTR
006614 202000 002161*   MOVEM A,CT33
006615 550040 002124*   HRRZ B,COM
006616 260740 006604*   TSX QCTR
006617 202000 002162*   MOVEM A,CT34
006620 550040 002130*   HRRZ B,CK
006621 260740 006604*   TSX QCTR
006622 202000 002163*   MOVEM A,CT35
006623 200000 002213*   300070
006624 221000 023420    300080
006625 230000 000042*   300090
006626 202000 002212*   301000
006627 201100 000001    301100
006630 200002 002176*   301200
006631 231000 000010    301300
006632 221000 000002    301400
006633 272002 002176*   301500
006634 365100 006630*   301600
006635 200040 002165*   301700
006636 221040 001750    301800
006637 231040 000002    301900
006640 230040 002166*   302000
006641 264000 004262*   302100
006642 332000 002166*   302200
006643 350002 002414*   302300
006644 263740 000000    302400

QCTR:      OCT GR,CT33; COUNT GREEN QUEUE
          COUNT COMPUTE QUEUE
          COUNT CK,CT35; COUNT WAIT FOR BUFFER QUEUE
          COUNT COMPUTE QUEUE

MOVE      A,CT44;
          A,D10000;
          A,TIME
          IDIV
          IMULI
          IDIV
          MOVE
          A,CT43;
          A,CT27A (C)
          IDIVI
          A,D8
          IMULI
          A,2
          ADDM
          A,CT27A (C);
          CT27A AND CT27B
          SOJGE
          C,GT1;
          MOVE
          B,CT13;
          IMULI
          B,D1000
          IDIVI
          B,2;
          DISPLAY AT 2 MS.
          IDIV
          B,CT14;
          JSR
          CVTL
          SKIP
          CT14;
          AOS
          T5 (C);
          DONE

          MAKE LOG PRINT OCTAL STATION #
          1/10THS OF MS./STATEMENT
          1/10THS OF MS./STATEMENT
          SKIP IF NO STATEMENTS INTERPRETED
          RECORD IN DISTRIBUTION

```

```
30260 ; USES REGISTERS A,B,C,D,E,F,G. S MUST CONTAIN CONTEXT FOR OUTPUT
006645 200140 010567*      30270 DISINT: MOVE D,[XWD -N,S,0]; WELL CHECK ALL STATIONS
006646 201300 000010      30280           MOVEI G,1D8; AND OUTPUT 8 PER LINE
006647 260740 010422*      30290           TSX CMESS
006650 260740 006653*      30300           TSX D12;MAKE UP INITIALS LINE
006651 321140 006646*      30310           JUMPL D,DI1; BACK AROUND IF MORE STATIONS
006652 263740 000000      30320           DONE
006653 336243 002572*      30340 DIS2: SKIPN F,MINT(D); SKIP IF INITIALS ARE THERE
006654 254000 006661*      30350           JRST D13;
006655 550000 000003      30360           HRRZ A,D
006656 260740 006700*      30370           TSX OCT;
30380           XMT 5,F; CONVERT STA # TO OUTPUT
30390           OPDEF X[5B12] AND THE INITIALS
006657 000240 000005      X,F
006660 370000 000006      30400 SOS G; COUNT INITIALS IN BUFFER
006661 252140 006663*      30410 AOBJP D,DI4; JUMP IF WE HAVE FINISHED THE TABLE
006662 327300 006653*      30420           JUMPG G,DI2;
006663 263740 000000      30430           DONE
30430           DI4:
```

30450	*	DISPLAY LOG-ONS AND LOG-OFFS IF PSEUDO SWITCH DOAF IS SET	
30460	*	D CONTAINS "ON-" OR "OFF-" DESTROYS A,B,C,E,S.	
30470	;		
30480			
30490	DOF:	GET THE SWITCHES	
006664	200000 006364*	MOVE A,SWITCH	
006665	606000 000010	TRNN A,DOAF;	SKIP IF DISPLAY INDICATED
006666	263740 000000	DONE	;FORGET IT IF NONE
006667	550240 000016	HRNZ F,S	
006670	260740 010422*	30520 CMESS	
006671	260740 006673*	30530 TSX DOF1;	MAKE UP LOG ON/OFF LINE
006672	263740 000000	30540 TSX DONE	
		30550	
		30560	
		30570 DOF1: XMT 4,D; "ON" OR "OFF"	
006673	000200 000003	X D OPDEF X [4B12]	
006674	550000 000005	30580 HRRZ A,F	
006675	260740 006700*	30590 TSX OCT;	
		30600 XMT 5,INITIALS	
006676	000240 004517*	X INITIALS OPDEF X [5B12]	
006677	263740 000000	30610 DONE	
		30620	
		30630	
		30640	
		30650 CONVERT STATION # TO OCTAL ASCII (TWO DIGITS ONLY)	
006700	260740 006703*	30660 OCT: TSX OCT1	
006701	010000 000055	30670 INS "	
006702	263740 000000	30680 DONE	
		30690	
006703	231000 000010	30700 OCT1: IDIVI A,10	
006704	435000 000060	30710 ORI A,60	
006705	136000 000002	30720 IDPB A,C;	
006706	435040 000060	30730 ORI B,60	
006707	136040 000002	30740 IDPB B,C	
006710	263740 000000	30750 DONE	

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
DISPLAY STATISTICAL DISTRIBUTIONS

PAGE 97

006711	201240	000020	30770	ODIS:	MOVEI	F,STAT2;	DISPLAY TABLE SIZE
006712	200445	006716*	30780		MOVE	J,STAT1 (F) ;	STAT TABLE POINTER
006713	260740	007057*	30790		TSX	TABO;	FORMAT AND OUTPUT STAT LINE
006714	365240	006712*	30800		SOJGE	F,--2	
006715	263740	000000	30810	DONE			
			30820				
			30830	STAT1:	XWD	0,T11;	T11, T10, AND T9 ARE USED
			30840		XWD	0,T10;	BY JOE TO RECORD COUNTS OF EXECUTION
			30850		XWD	0,T9;	BY VERB TYPE
			30860		XWD	0,TINT;	TOTAL I/O CHARACTERS
			30870		XWD	0,OINT;	OUTPUT INTERRUPTS
			30880		XWD	0,IINT;	INPUT INTERRUPTS
			30890		XWD	0,CPOL;	CHAR/OUTPUT LINE
			30900		XWD	0,CPIL;	CHAR/INPUT LINE
			30910		XWD	0,DSIZE;	SIZE
			30920		XWD	0,DCPI;	COMPUTE TIME PER INTERACTION
			30930		XWD	0,DTIM;	INTERACTION TIME
			30940		XWD	0,CCTIM;	COMPUTE TIME
			30950		XWD	0,SESTIM;	SESSION TIME
			30960		XWD	0,T5;	INTERPRETATION RATE
			30970		XWD	0,T8;	TASK TURN-AROUND TIME
			30980		XWD	0,T7;	MISC. STATISTICS
			30990		XWD	0,T6;	GENERAL COUNTS AND BLOCKSIZE DISTRIB.
			31000				
			31010	STAT2=-STAT1-1			
			31020				

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
COUNT TIME AND GATHER STATISTICS

PAGE 98

006737	200040	000033*	31040	STAT:	MOVE	B, SEC
006740	306040	000036	31050		CAIN	B,D30
006741	260740	003532*	31060		TSX	TRST;
006742	275040	000074	31070		SUBI	B,D60
006743	321040	007135*	31080		JUMPL	B,STR2,6;
			31090	:		DONE IF NOT A NEW MINUTE
			31100	;		NEW MINUTE
006744	571700	777777	31110		HRREI	S,-1;
006745	202040	000033*	31130		MOVEM	B,SEC
006746	260740	007212*	31140		TSX	UP;
006747	350040	000032*	31150		AOS	B,MIN;
006750	231040	000017	31160		IDIVI	B,K3;
006751	200140	006664*	31170		FSW D	
006752	606140	000002	31180	MOVE D,SWITCH	TRNN	D,DUL;
006753	336000	000002	31190		SKIPN	C;
006754	260740	006645*	31200		TSX	DISINT
006755	200140	006751*	31210	MOVE D,SWITCH	FSW D	
006756	626140	000020	31220		TRNN	D,DSTAT;
006757	254000	006762*	31230		JRST	STA3
006760	202140	006755*	31240		MOVEM	D,SWITCH
006761	260740	006711*	31250		TSX	ODIS;
006762	200140	006760*	31260	STA 3: FSW D		DISPLAY STATISTICS
006763	606140	000100	31270	MOVE D,SWITCH	TRNN	D,OISW;
006764	254000	006775*	31280		JRST	STA1
006765	201240	000021	31290		MOVEI	F,STAT2+1
006766	135440	004140*	31300		LDB	J,C5
006767	303440	007043*	31310		CALLE	J,STA2
006770	254000	006775*	31320		JRST	STA1
006771	322440	006775*	31330		JUMPE	J,STA1
006772	274240	000011	31340		SUB	F,J
006773	200445	006716*	31350		MOVE	J,STAT1 (F)
006774	260740	007057*	31360		TSX	TABO

```

006775 200140 010544*    31380   STA1:      MOVEI    B,4
006776 201040 000004     31390   MOVEI    B,4
006777 350000 002426*    31400   AOS      T6
007000 134100 000003     31410   ILDB    C,D
007001 272101 002433*    31420   ADDM   C,T6+5 (B)
007002 367040 007000*    31430   SOJC   B,-2
007003 200040 002204*    31440   MOVE   B,CT18
007004 272040 002433*    31450   ADDM   B,T6+5;
007005 200040 002207*    31460   MOVE   B,CT19
007006 272040 002432*    31470   ADDM   B,T6+4;
007007 200040 002203*    31480   MOVE   B,CT19A
007010 272040 002431*    31490   ADDM   B,T6+3;
007011 200040 002166*    31500   MOVE   B,CT14
007012 272040 002430*    31510   ADDM   B,T6+2;
007013 200040 002165*    31520   MOVE   B,CT13
007014 272040 002427*    31530   ADDM   B,T6+1;
007015 200040 002205*    31540   MOVE   B,CT18A
007016 272040 002440*    31550   ADDM   B,T7;
007017 332000 002201*    31560   SKIPE CT11;
007020 254000 007025*    31570   JRST   STA1,5
007021 350000 002447*    31580   AOS    T7+7;
007022 201040 007020    31590   MOVEI  B,*D3600
007023 274040 002165*    31600   SUB    B,CT13;
007024 272040 002450*    31610   ADDM   B,T7+8;
007025 200040 000032*    31620   MOVE   B,MIN
007026 275040 000074     31630   SUBI   B,*D60
007027 321040 007114*    31640   JUMPL B,ST1;
007027 321040 007114*    31650   ;      NEW HOUR
007027 321040 007114*    31660   ;
007030 402000 002210*    31670   SETZM CT17
007031 402000 002211*    31690   SETZM CT23;
007032 202040 000032*    31700   MOVEM B,MIN;
007033 350040 000031*    31720   AOS    B,HR
007034 606040 000003     31730   TRNN  B,3;
007035 260740 006711*    31740   ODIS   B,HR
007036 200040 000031*    31750   MOVE   B,HR
007037 275040 000030     31760   SUBI   B,*D24
007040 321040 007043*    31770   JUMPL B,STA2;
007041 202040 000031*    31780   MOVEM B,HR
007042 260740 007162*    31790   TSX    UPDATE;
007043 260740 007100*    31800   STA2: TSX    HMES;
007044 254000 007114*    31810   JRST   ST1

```

			HANDLES ALL TAPE OUTPUT EXCEPT ACCOUNTING RECORDS
007045	260740 010343*	31830	
007046	263740 000000	31840	TSX GETBUF
007047	504140 000001	31860	DONE
007050	544141 000000	31870	HRL D,B
007051	275200 000002	31880	HLR D,(B);
007052	505200 000003	31890	SUBI E,2;
007053	260740 010433*	31900	HRLI E,D;
007054	271200 000002	31910	TSX B,9,1;
007055	260740 010305*	31920	ADDI MOVE WORDS TO BUFFER
007056	263740 000000	31930	E,2; RESTORE
		31940	PUTB; START OUTPUT
		31950	DONE

		31970	:	OUTPUTS TABLE OF SIZE 10 FROM (J) , USES LOTS OF REGISTERS
007057	260740 010422*	31980		
007060	260740 007062*	31990	TABO:	TSX CMESS
007061	263740 000000	32010		TSX TABO1;
		32020	DONE	FORMAT THE LINE
007062	571340 777766	32030	TABO1:	HRREI H,-12
007063	201300 000006	32040	ST. 1:	MOVEI G,6;
007064	201400 000040	32050		MOVEI I,"";
007065	402000 000003	32060		SETZM D;
007066	200011 000000	32070		MOVE A,0 (J) ;
007067	350000 000011	32080		AOS J
007070	260740 007240*	32090		TSX FRMT;
007071	346340 007063*	32100		AOJN H,ST.1
		32110		XMT 5,ST.12
			OPDEF X [5B12]	
007072	000240 007077*	X ST,-12		
007073	201000 000021	32120		MOVEI A,STAT2+1
007074	274000 000005	32130		SUB A,F
007075	260740 006403*	32140		CTD;
007076	263740 000000	32150		DONE
007077	201004 021500	32160		
		32170	ST. 12:	ASCII ? * ?

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.

PAGE 102

OUTPUT HOURLY HEADING LINE

007100	260740	010422*	32190	HMES:	TSX	CMESS
007101	260740	007103*	32200		TSX	HMESS1
007102	263740	000000	32210		DONE	
007103	010000	000012	32220	HMES1:	INS	12;
007104	200000	000031*	32230		MOVE	A,HR
007105	260740	006403*	32240		TSX	CTD;
007106	010000	000072	32250		INS	" :
007107	010000	000040	32260		INS	" "
			32270			
			32280			
				OPDEF X [10B12]	XMT 10,DATE	
007110	000400	000023*	X DATE			
			32290		XMT 2,CRLF	
007111	000100	004134*	OPDEF X [2B12]			
			X CRLF			
007112	004400	002231*	32300	XMT' 110,HEAD;		72 HEADING CHARACTERS
007113	263740	000000	OPDEF X [110B12]			
			X HEAD			
			32310	DONE		

OUTPUT DETAIL STATISTICS ON THE MINUTE

```

007114 260740 006612*   32330 ST1:      TSX      COMPUTE NEW COUNTS
007115 332000 000032*   32340 ST1B:     SKIPE    MIN
007116 254000 007121*   32350 JRST     *+3
007117 402000 000042*   32360 SETZM    TIME;
007120 402000 002213*   32370 CT44
007121 200040 002174*   32380 MOVE    B,CT25
007122 270040 002175*   32390 ADD     B,CT26
007123 270040 002176*   32400 ADD     B,CT27A
007124 270040 002177*   32410 ADD     B,CT27B
007125 322040 007130*   32420 JUMPE   B,ST2.5-1;
007126 260740 010422*   32430 TSX      NO OUTPUT IF NO ERRORS OR CHAR.
007127 260740 007136*   32440 CMESS   FORMAT TO BUFFER
007130 571040 777745   32450 STX     ST1A;
007131 200201 002745*   32460 HRREI   B,-CTPARS
007132 607200 400000   32470 MOVE    E,CTPAR+CTPARS (H); GET PARAMETER WORD
007133 402004 000000   32480 TLNN    E,400000; SKIP IF SHOULD NOT ZERO THE COUNTER
007134 341040 007131*   32490 SETZM   (E)
007135 263740 000000   32500 AOJL    B,ST2.5
                               DONE

32510 ;           FORMAT A LINE FOR THE CONSOLE TTY
32520 ;           FORMAT A LINE FOR THE CONSOLE TTY
32530 ;           FORMAT A LINE FOR THE CONSOLE TTY
32540 ST1A:      MOVE   A,MIN   CONVERT MINUTES
32550 TSX       CTD;    H,-CTPARS; TABLE SIZE
32560 HRREI   H," ";   SET ALTERNATE OUTPUT CHARACTER
32570 MOVEI   I," ";   SETZM   D;
32580 MOVEI   I," ";
32590 LDB     G,[POINT 5,CTPAR+CTPARS (H),5]; GET FIELD SIZE
32600 LDB     F,[POINT 12,CTPAR+CTPARS (H),17]; GET SCALE FACTOR
32610 MOVE   B,CTPAR+CTPARS (H)
32620 MOVE   A,(B);   MOVEI   A,*+3
32630 JUMPN   D,*";   JRST    ST2.4
32640 MOVEI   D,*";   IDIV    A,F;
32650 JRST    A,F;   JUMPN   A,ST2.4;
32660 IDIV    A,B;   MOVE   A,B
32670 JUMPN   IMULI  A,*D10;
32680 MOVEI   D,*";   MOVEI   D,*";
32690 TSX     FRMT;   TSX     H,ST2;
32700 AOJL    DONE
32710 ST2.4:    ;           MAKE A PLACE FOR THE *
32720 ;           MAY HELP BUT IT AINT PERFECT
32730 ;           FORMAT THE OUTPUT LINE
32740 ;           GO AROUND IF MORE COUNTERS

```

```

ROUTINE TO INCREMENT AND UPDATE DATE.

        B, DAY;          UPDATE: AOS      MOVE
        D, [POINT 5, MONS]    D      MOVE
        C, MONTH           C      MOVE
        D      IBPJ
        D      SJIN
        C,-1;      MOVE
        C, MONTH           C      MOVE
        A,D      LDB
        C,2      CAIE
        D      JRST
        UD2;      NOT FEBRUARY
        C, YEAR           MOVE
        C, 3;      MOVE
        AOSA
        A;      SUB
        B,A      JUMBLE
        B, UP1;
        AOS      NOT A NEW MONTH
        B, MONTH;      BUMP THE MONTH
        A,1      MOVEI
        A, DAY;      MOVEM
        TSX      SET DAY = 1
        MDS;      TIME FOR DISC ACCOUNTING
        MOVE
        SUBI
        B, D12
        B, UP1;
        AOS      NOT A NEW YEAR
        YEAR
        MOVEI
        A, 1      MOVEM
        A, MONTH;      SET MONTH = 1

        CONVERT DATE AND TIME TO ASCII

        MOVE
        A, BLANKS
        A, DATE
        A, DATE + 1;      BLANK THE DATE
        MOVEM
        A, ATIM;
        AND TIME
        C, [POINT 7, DATE]
        MOVE
        A, MONTH
        MOVE
        TSX
        CTDB
        INS
        " / "
        MOVE
        A, DAY
        TSX
        CTDB
        INS
        " / "
        MOVE
        A, YEAR
        TSX
        CTDB
        C, [POINT 7, ATIM]
        MOVE
        A, HR
        MOVE
        A, MIN
        TSX
        CTDB;
        " : "
        MOVE
        A, MIN
        TSX
        CTDB;
        DONE

        ADATE=UP 1      RADIX
        MONS:      BYTE
        33240      33250      33260      33270
        007236    776377 577676
        007237    773777 574000
        10      (5) 31, 28, 31, 30, 31, 31, 30, 31, 30, 31, 30, 31

```

```

33290 ; A CONTAINS SCALED COUNTER VALUE, C THE POINTER TO THE
33300 ; OUTPUT LINE, G THE FIELD SIZE, AND I THE ALTERNATE
33310 ; OUTPUT CHARACTER.
33320 ; D, B ARE CLOBBERED.

007240 315006 000000 FRMT: CAMGE A,T80 (G); TOO BIG FOR FIELD?
007241 254000 007246* 33350 JRST ST4; NO
007242 201140 000041 33360 MOVEI D,"!"; YES, SET OVERPUNCH
007243 231000 000012 33370 IDIVI A,*D10;
007244 311006 007240* 33380 CAML A,T80 (G);
007245 344140 007243* 33390 AQUA D,ST3;
007246 306300 00001 CAIN G,1 IF STILL TOO BIG:
007247 201400 000060 33400 MOVEI I,"0";
007250 230006 000000 33410 IDIV A,T80,99 (G)
007251 326000 007254* 33420 SET ALTERNATE TO ZERO ON LAST COLUMN
007252 200000 000010 33430 JUMPN A,ST5
007253 254000 007256* 33440 MOVE A,I;
007254 201400 000060 33450 JRST ST6 IF ZERO GET ALTERNATE
007255 435000 000060 33460 MOVEI I,"0";
007256 136000 000002 33470 ORI A,60;
007257 250000 000001 33480 IDPB A,C;
007258 367300 007246* 33490 EXCH A,A+1;
007260 322140 007263* 33500 SOIG G,ST4;
007261 137140 000002 33510 JUMPE D,ST7;
007262 263740 000000 33520 DPB D,C;
007263 ; DONE

```

	33550	;	MASTER MONITOR ENTRY POINT
007264 402000 000043*	33560	SIGR: SETZM	T.CU; KEEP IT FROM COUNTING TOO HIGH
	33570	SIGR1=SIGPR	
	33580	FSW B;	GET DATA SWITCHES
007265 200040 006762*	MOVE B,SWITCH		
007266 602040 200000	33600	TRNE B,SDS	GO TO SHUTDOWN PROCEDURE
007267 260740 006104*	33610	TSX SDP;	
007270 602040 100000	33620	TRNE B,OFFS	GO TO BLAST OFF PROCEDURE
007271 260740 005512*	33630	TSX BOFF;	
007272 332000 003162*	33640	SKIPE CKER	
007273 260740 005613*	33650	TSX GRONK	SKIP IF NO DISC RESTART WAITING
007274 332700 002126*	33660	SKIPE S,DCT;	TRY THE RESTART
007275 260740 004642*	33670	TSX DSTRAT;	COMPLETION SIGNAL FROM DISC?
007276 332140 006061*	33680	SKIPE D,DISC,S;	KEEP TABS ON THE DISC
007277 260740 005573*	33690	TSX DISCP;	
007300 332000 006102*	33700	SKIPE SKT	TIME FOR NEXT SKULK RECORD
007301 260740 006007*	33710	TSX MDS;	
007302 332000 000115*	33720	SKIPE SG,L	
007303 260740 006134*	33730	TSX PRSIG;	PROCESS SIGNALS FROM DISTRIBUTOR
007304 260740 006737*	33740	TSX STAT;	DO STATISTICS
007305 332700 002134*	33750	SKIPE S,ABG	
007306 260740 005543*	33760	TSX MSGPR;	GET A BUFFER FOR AWAITING GREEN
007307 332700 002135*	33770	SKIPE S,QP	
007310 260740 005561*	33780	TSX PQP;	PROCESS THE PAUSE QUEUE
007311 332000 002137*	33790	TSX QM	
007312 260740 006304*	33800	TSX OF2;	TRY TO ENABLE USER IN THE QUEUE
007313 332000 005471*	33810	SKIPE SS99;	SKIP IF NO COMPACT FOR IN SWAP
007314 260740 005456*	33820	TSX SS90;	GO COMPACT
007315 332700 002136*	33830	SKIPE S,QDM	
007316 260740 007322*	33840	TSX DM90;	TAKE CARE OF SWAPS TO INCREASE CORE
007317 336000 002107*	33850	SKIPE DMBY	
007320 260740 005211*	33860	TSX SELSWP;	
007321 254000 007335*	33870	JRST SELINT;	SELECT FOR SWAP
	33880	;	GO TO SELECT FOR INTERPRET
	33890	;	PROCESS THE QUEUE FOR TRANSFER TO DRJM
007322 332000 002107*	33900	DM90: SKIPE DMBY	
007323 263740 000000	33920	DONE	
007324 275700 000141*	33930	SURI S,S,Q	PRIORITY TO COME IN
007325 020000 000006	33940	CHS QC,S;	
007326 135100 000264*	33950	LDB C,S,BLOCK	
007327 275100 000001	33960	SUBI C,1;	BLOCKS TO GO OUT
007330 476000 003331*	33970	SETOM DMIN;	INDICATE NO IN REQUESTED
007331 260740 003234*	33980	TSX OSWAP;	START HIM OUT
007332 263740 000000	33990	DONE	
	34000		
	34010		

		TABLE OF STATES WHICH ARE SEARCHED TO FIND A USER	
		SI1:	BYTE
007333	000420 620514	34050	(5) TOF.S,ON.S,RC.S,RI.S,RIB.S,UC.S,QC.S,COM.S,END.S
007334	177400 000000	34060	
007335	201440 000010	34070	SELINT: MOVEI J,10; COUNT TO SAVE US FROM THE MACHINE
007336	200040 010554*	34080	MOVE B,IPOINT 5,SI1]; GET TABLE POINTER
007337	134100 000001	34090	IILDB C,B; INDEX TO STATE HEADER
007339	306100 000037	34100	CAIN C,END.S; SKIP IF NOT END
007340		34110	SI4; END OF SEARCH
007341	254000 007360*	34120	SOSGE J
007342	375000 000011	34130	HALT 30; CAIN OR ILDB HAS FAILED
007343	040000 000030	34140	H30: HRRZ C,S,QUE (C); LIST HEADER
007344	550102 002115*	34150	JRST C,SI12; JUMP IF NO ENTRY
007345	322100 007337*	34160	
007346	550700 000002	34170	HRRZ S,C STATION INDEX
007347	275700 000141*	34180	SUBI S,S,Q; GET CORE CELL
007350	135140 000263*	34190	LDB D,S,COR; GET STATE
007351	135240 000261*	34200	LDB F,S,STA; FOUND ONE; JUMP TO INITIALIZE
007352	326140 007406*	34210	JUMPN D,INTINT; IS IT ON?
007353	306240 000001	34220	CAIN F,ON,S; JRST SI7; YES, GO TO FIND CORE
007354	254000 007367*	34230	C,0 (C); NOT IN SO LOOK FURTHER
007355	550102 000000	34240	JUMPN C,SI13; END OF LIST - GO FOR NEXT LIST
007356	326100 007346*	34250	JRST SI2;
007357	254000 007337*	34260	
007360	336000 002107*	34270	MONITOR IDLE LOOP
007361	354000 002201*	34280	
007362	336000 002124*	34290	CANT FIND WORK; IS DRUM BUSY?
007363	254000 007264*	34300	NO WORK ADD TO IDLE TIME.
007364	350000 002200*	34310	AOSA CT11; COM
007365	350000 002441*	34320	SKIPN JRST SIGPR;
007366	254000 007264*	34330	34340 CT12; NOT UNOVERLAPPED IF NO COMPUTE
		34350	AOS T7*1 UNOVERLAPPED TICKS
		34360	JRST SIGPR; GO PROCESS SIGNALS
		34370	
		34380	
		34390	

34410 ; PROCESS AN "ON" USER
007367 201140 0000000 34420
007370 336003 000272* 34430
007371 254000 007376* 34440
007372 350000 000003 34450
007373 311140 000313* 34460
007374 254000 007355* 34470
007375 254000 007370* 34480
007376 200003 000272* 34490
007377 661000 4000000 34500
007400 202003 000272* 34520
007401 200040 000003 34530
007402 137700 000317* 34540
007403 271140 000020 34550
007404 137140 000263* 34560
007405 350000 002202* 34570
; NOPE
SI7: MOVEI D,0
SKI PN CORE (D)
JRST SI8;
AOS D
HAVE WE SEARCHED ALL
YES, THERE ARE NONE AVAILABLE
NO, GO FOR NEXT
SI8+1;
CAML D,N,C;
JRST SI3..5;
JRST SI7+1;
MOVE A,CORE (D)
TLO A,400000;
MOVEM A,CORE (D)
MOVE B,D
DPB S,S,UR;
ADD I D,BBLOCK;
DPB D,S,COR;
AOS CT39;

34600	*	ON ENTRY F CONTAINS STATE, S THE SELECTED USER, AND D HIS CORE LOCATION
34610	,	
34620		
007406	275140 000020	INTINT: SUBI D,BBLOCK
007407	200143 000272*	MOVE D,CORE(D)
007410	603140 200000	TLNE D,200000
007411	2540000 007355*	JRST SI3..5;
007412	331000 000016	SKIPL S
007413	303700 000050	CAILLE S,N,S;
007414	040000 000001	HALT 1;
007415	200305 002141*	MOVE G,T10..(F);
007416	607300 000004	TLNN G,BUFBIT;
007417	254000 007423*	JRST INT2
007420	260740 010343*	TSX GETBUF
007421	254000 007355*	JRST SI3..5;
007422	200000 000004	MOVE A,E;
007423	260740 005110*	TSX SRR;
007424	603300 000040	TLNE G,ONBIT;
007425	254000 007431*	JRST INT3;
007426	200040 006676*	MOVE B,INITIALS
007427	312056 002572*	CAME B,MINT(S);
007430	254000 007456*	JRST KILL1;
007431	603300 000002	TLNE G,INBIT;
007432	350000 004402*	AOS RISIG;
007433	135040 00266*	LDB B,S,INR
007434	436040 007432*	ORM B,RISIG;
007435	402000 000001	SETZM B
007436	137040 000266*	DPB B,S,INR;
007437	135040 000267*	LDB B,S,DU
007440	332000 000001	SKIPE B
007441	476000 004614*	SETOM SPARE4;
007442	607300 000001	TLNN G,RCBIT;
007443	254000 007446*	JRST INT1
007444	550016 000331*	HRRZ A,S,BUF(S);
007445	402016 000331*	SETZM S,BUF(S);
007446	020000 000010	CHS CU,S;
007447	200200 000000	MOVE E,A;
007450	135040 000262*	LDB B,S,TM
007451	202040 000043	MOVEM B,T,CU;
007452	201040 000000	MOVEI TLNE G,CORBIT
007453	603300 000020	MOVEI B,1;
007454	201040 000001	JRST MONE1;
007455	254000 004351*	35030
007456	202700 000021*	KILL1: MOVEV S,CUI;
007457	202056 002642*	MOVEV B,SUM(S);
007460	201000 000073	MOVEI A,73
007461	254000 004162*	KILL;

```

35090          JRST:      *+1
                MOVE      D,[POINT 6,CT48A]
                ILDB      B,D
                ADDI      B,1
                DBB      B,D;
                COUNT IN BLOCK SIZE DISTRIBUTION

BJ:           MOVEI     B,NEXT
                MOVEM     B,INTENT;
                MOVE     C,CUI
                MOVE     B,ASCII /??
                MOVEM     B,MINT(C);
                MOVEM     B,INITIALS
                MOVE     B,BLANKS
                MOVEM     B,JOBNO
                SETZM    COMTIM
                SETZM    PAGNO
                SETZM    SPARE1;
                SETZM    SPARE2;
                SETZM    SPARE3;
                EXCH     PP,PPSAV
                SETZM    SEQ;
                SETZM    ISEC
                MOVEM     B,ONTIME;
                MOVE     S,CUI
                HRLZM    B,USTAT(S);
                MOVEI     B,1;
                JRST     BJLEV
                "JOSS HERE *** INITIALS PLEASE"
                INCREMENT SEQUENCER

BJ1:          SEND     BJM1;
                AOS      SEQ;
                JRST     BJSU
                GO TO APPROVE INITIALS
                NO GOOD
                GOOD - "JOB # PLEASE"
                INCREMENT SEQUENCER

BJ2:          TSX      BJA1;
                JRST     BJ2-1;
                SEND     BJM2;
                AOS      SEQ;
                JRST     BJSU
                "INITIALS AGAIN"

BJ2-1:        SEND     BJM3;
                JRST     BJSU

```

```

007526 260740 007656*      35500          BJ3:      TSX      BJAS;      GO TO APPROVE JOB #
007527 254000 007536*      35520          JRST      AOS      SEQ;      NO GOOD
007530 350000 007522*      35530          AOS      SEQ;
007531 350000 007530*      35540          SEQ;      A,D1000;
007532 301000 001750      35550          CALL      BJ6;      GOOD - INCREMENT SEQUENCER
007533 254000 007567*      35560          JRST      MOVEI   B,2;
007534 201040 000002      35570          JRST      BULEV;  SKIP IF NOT AN RPN
007535 254000 007546*      35580          JRST      "ARE YOU PLAYING?" YES, GO GET DEPARTMENT
                                         RETURN BUFFER
                                         RETURN WILL BE TO INTERP. FIRST ENTRY
                                         "JOB # AGAIN"
007536 030000 010472*      35600          SEND      BJM4;
007537 350000 007531*      35610          AOS      SEQ;
007540 254000 007545*      35620          JRST      BJSU
                                         GO TO APPROVE JOB #
007541 260740 007656*      35630          BJ4:      TSX      BJAS;      JOB # OK
007542 254000 007544*      35640          JRST      *+2;
007543 254000 007531*      35650          JRST      BJ3,*1;
007544 030000 010502*      35660          SEND      BJM5;
007545 201040 000004      35670          MOVEI   B,4;
007546 250740 000020*      35680          EXCH      PP,PPSAV
007547 254000 004304*      35690          MONIT;  PPSAV
                                         JRST      GO TO MONITOR
                                         GET THE SEQUENCER
007550 550040 007537*      35710          NEXT:    HRRZ      B,SEQ;
007551 250740 000020*      35720          EXCH      PP,PPSAV
007552 254021 007553*      35730          JRST      @BJSW(B);
                                         DISPATCH TO PROPER SEQUENCE
                                         GO TO APPROVE JOB #
007553 000000 007514*      35740          BJ5W:    XWD      0,BJ1
007554 000000 007517*      35750          XWD      0,BJ2
007555 000000 007526*      35760          XWD      0,BJ3
007556 000000 007541*      35770          XWD      0,BJ4
007557 000000 007562*      35780          XWD      0,BJ5
007558 000000 007572*      35790          XWD      0,BJ6
007561 000000 007601*      35800          XWD      0,BJ7
                                         35810          XWD      0,BJ8
                                         35820          XWD      0,BJ9
                                         35830

```

007562	200700	000021*	35850	BJ5:	MOVE	S, CUI:	RECOVER STATION #
007563	200140	010576*	35860		MOVE	D, [ASCII ? ON-?]	DISPLAY INITIALS ON TTY
007564	260740	006664*	35870		TSX	DOF;	
007565	250740	000020*	35880		EXCH	PP, PPSAV	
007566	254000	004155*	35890		INTBEG;		GO TO BEGINNING OF INTERPRETER
007567	350000	007550*	35910	AOS	SEQ		
007570	030000	010520*	35920	SEND	BJM7;	"DEPARTMENT"	
007571	254000	007545*	35930	JRST	BJSU		
007572	260740	007763*	35940				
007573	254000	007576*	35950	BJ7:	TSX	BJAD;	GO TO APPROVE DEPT
007574	370000	007567*	35960		JRST	BJ8;	NO GOOD
007575	254000	007534*	35970		SOS	SEQ;	OK EXIT
			35980		JRST	BJ3_3	
007576	350000	007574*	35990				
007577	030000	010524*	36000	AOS	SEQ	"DEPT NAME OR NUMBER"	
007600	254000	007545*	36010	SEND	BJM8;		
			36020	JRST	BJSU		
007601	260740	007763*	36030				
007602	254000	007577*	36040	BJ9:	TSX	BJAD;	TRY FOR APPROVAL AGAIN
007603	370000	007576*	36050		JRST	BJ8+1;	NO GOOD = KEEP AT HIM
007604	254000	007574*	36060		SOS	SEQ	
			36070		JRST	BJ7+2;	TAKE OK EXIT

```

007605 200040 004130*    36090      BJA1:      MOVE     B, BLANKS
007606 202040 007474*    36100      MOVEM   B, INITIALS
007607 201400 000004     36110      MOVEI   I, 4;
007610 201100 007632*    36120      MOVEI   C,SCT1;
007611 200040 004136*    36130      MOVE    B,C3;
007612 260740 010103*    36140      TSX    SCAN;
007613 306400 000004     36150      CAIN   I, 4;
007614 254000 007627*    36160      JRST   B,JA4
007615 350017 000000     36170      AOS    0 (PP);
007616 200040 007606*    36180      MOVE   B, INITIALS
007617 200100 000021*    36190      MOVE   C,CUI
007620 202042 002572*    36200      MOVEM B,MINT (C);
007621 263740 000000     36210      DONE
007622 262740 000000     36220      POP    PP, 0;
007623 263740 000000     36230      POP    DONE;
007624 371000 000010     36240      NOT APPROVED EXIT FOR INITIALS SCAN
007625 254003 000003     36250      ALPHA1: SOSL   I;
                                         3 (D) ;
                                         36260      MOVE   B,INITIALS
                                         36270      MOVEM B,INITIALS
                                         36280      NOT APPROVED EXIT FOR INITIALS SCAN
                                         36290      POP    PP, 0
                                         36300      BJA3:  MOVE   B, [ASCII /???
                                         36310      BJA4:  MOVEM B,INITIALS;
                                         36320      36330      /] RESTORE TEST CELL (FOR GOOD DRUM DATA)
                                         36340      DONE; TAKE NOT APPROVED EXIT

```

007632	255000	000000	36360	36370	SCT1:	NOP;	
007633	265140	007624*	36380	36390	JSP	D,ALPH1;	BLANK
007634	254000	007626*	36400	36410	JRST	BJA3;	ALPHA
007635	254000	007626*	36400	36410	JRST	BJA3;	NUMERIC
007636	254000	007626*	36400	36410	JRST	BJA3;	SPECIAL
007637	255000	000000	36420	36430	NOP;		ILLEGAL
007640	254000	007626*	36430	36440	JRST	BJA3;	PERIOD
007641	255000	000000	36440	36450	NOP;		TAB
007642	265140	007655*	36450	36460	JSP	D,END;	DASH
							END OF STRING
007643	405100	000017	36470	DIGIT:	ANDI	C,17;	MASK TO NUMERIC
007644	326000	007647*	36480	JUMPN	A,DIG1;	JUMP IF NOT TESTING FOR LEADING ZEROS	
007645	326100	007647*	36490	JUMPN	C,.+2;	IT IS NON-ZERO	
007646	254003	000000	36500	JRST	0(D);	IGNORE LEADING ZEROS	
007647	221000	000012	36510	DIG1:	IMULI	A, ¹ D10	
007650	270000	000002	36520	ADD	A,C;	CONSTRUCT BINARY RPN IN REG A	
007651	435100	000060	36530	ORI	C,60;	MAKE IT AN ASCII AGAIN	
007652	371000	000010	36540	SOSL	I;	COUNT CHARACTERS	
007653	254003	000003	36550	DASH:	3(D);	ADD TO STRING	
007654	254000	007622*	36560	JRST	BJA2;	MORE THAN 4 INITIALS, NOT APPROVED EXIT	
007655	254003	000002	36570	END:	2(D);	EXIT FROM SCAN	

115 50 158

```

36590      BJA5:      MOVE    B,BLANKS
007656    200040 004130* 36600      MOVEI   B,JBNO
007657    202040 007476* 36610      MOVEI   A,0;
007660    201000 000000 36620      MOVEI   I,4;
007661    201400 000004 36630      MOVEI   C,SCT2;
007662    201100 007752* 36640      MOVE   B,C4;
007663    200040 004137* 36650      TSX    SCAN;
007664    260740 010103* 36660      MOVEM  A,SPARE1;
007665    202000 007501* 36670      JUMPE A,BJAN;
007666    322000 007723* 36680      CAIG   A,4;
007667    307000 000004 36690      JRST   BJAY;
007670    254000 007722* 36700      MOVEI   B,RPN5;
007671    201040 000011 36710      D,RPN-1(B);
007672    554141 007740* 36720      CAMG   A,D;
007673    315000 000003 36730      JRST   .+4;
007674    254000 007700* 36740      HRRZ   D,RPN-1(B);
007675    550141 007740* 36750      A,D   HIGH END OF OK RANGE
007676    317000 000003 36760      CAMG   BJAY;
007677    254000 007722* 36770      JRST   ITS OK
007700    367040 007640 36780      SOJG   B,BJA5,2;
007701    201040 000011 36790      SEARCH SPECIAL TABLE
007702    135140 007711* 36800      MOVEI   B,SRPNS
007703    322140 007711* 36810      LDB    D,PLOW
007704    315000 000003 36820      BJS1:   D,BJS2;
007705    254000 007711* 36830      JUMPE A,D;
007706    135140 007740* 36840      CAMGE A,D;
007707    317000 000003 36850      JRST   BJS2;
007710    254000 007713* 36860      LDB    D,PHIG
007711    367040 007702* 36870      CAMG   A,D;
007712    254000 007723* 36880      JRST   BJS3
007713    135140 007736* 36890      SOJG   B,BJS1
007714    336000 000021* 36900      JRST   BJAN;
007715    254000 007720* 36910      CAME   D,CUI
007716    312140 000021* 36920      JRST   BJS4;
007717    254000 007723* 36930      LDB    CUI;
007720    135140 007735* 36940      SKIPN  D,CUI
007721    202140 007502* 37000      BJS4:   CAME   BJS4;
007722    350017 000000 37010      JRST   D,CUI
007723    263740 000000 37020      LDB    BJS4;
007723      37030      MOVEM  D,SPARE2;
007723      37040      AOS    0 (PP);
007723      37040      DONE
007713      36940      HERE WE HAVE A JN IN OK RANGE
007714      36950      BJS3:   LDB    D,PSTN;
007715      36960      BJS3:   SKIPN  CUI;
007716      36970      BJS3:   JRST   ZERO IS OK
007717      36980      BJS3:   CAME   NOT ALLOWED ON THIS STATION
007720      36990      BJS3:   JRST   BJS4;
007721      37000      BJS4:   LDB    D,PDEP
007722      37010      BJS4:   MOVEM  D,SPARE2;
007723      37020      BJA5:   AOS    0 (PP);
007723      37030      BJA5:   DONE
007723      37040      HERE WE HAVE A JN IN OK RANGE

```

		SRPN# =	RADIX	TABLE OF SPECIAL RPNS	
007724	234027 555716	37060	BYTE	(8) 78, 105	(10) 950, 974;
007725	232023 411634	37070	BYTE	(8) 77, 104	(10) 900, 924;
007726	230017 101470	37080	BYTE	(8) 76, 103	(10) 800, 824;
007727	226012 571324	37100	BYTE	(8) 75, 102	(10) 700, 724;
007728	224212 261211	37110	BYTE	(8) 74, 1042	(10) 600, 649;
007729	222235 751045	37120	BYTE	(8) 73, 1047	(10) 500, 549;
007730	220215 440701	37130	BYTE	(8) 72, 1043	(10) 400, 449;
007731	216175 130617	37140	BYTE	(8) 71, 1037	(10) 300, 399;
007732	214154 620453	37150	BYTE	(8) 70, 1033	(10) 200, 299;
007733	211201 007723*	37160	RADIX	8	SACRAMENTO
007734	209120 007723*	37170	SRPN# = -SRPN;		
007735	341001 007723*	37180	TABLE SIZE		
007736	241001 007723*	37190	PDEP:	POINT	8, SRPN-1 (B), 7
007737	121201 007723*	37200	PSTN:	POINT	8, SRPN-1 (B), D15
007738	101201 007723*	37210	PLOW:	POINT	10, SRPN-1 (B), D25
007739	001201 007723*	37220	PHIG:	POINT	10, SRPN-1 (B), D35
		37230			
		37240			
		37250			
		37260			
		37270	RPN#:	RADIX	10
007741	023254 023417	37280	XWD	XWD	9900, 9999
007742	022745 023253	37290	XWD	XWD	9701, 9899
007743	022674 022674	37300	XWD	XWD	9660, 9660
007744	021640 022516	37310	XWD	XWD	9120, 9550
007745	020022 020034	37320	XWD	XWD	8210, 8220
007746	015530 015732	37330	XWD	XWD	7000, 7130
007747	015055 015100	37340	XWD	XWD	6701, 6720
007750	013114 013147	37350	XWD	XWD	5708, 5735
007751	001751 003410	37360	RPN# = -RPN;	XWD	1001, 1800
		37370	RPN# = -RPN;	RADIX	8
		37380			SIZE OF RPN TABLE

007752 255000 000000 37400
007753 254000 007622* 37410
007754 265140 007643* 37420
007755 254000 007622* 37430
007756 254000 007622* 37440
007757 255000 000000 37450
007760 254000 007622* 37460
007761 254000 007622* 37470
007762 265140 007655* 37480
 37490

SCT2: NOP; BJA2;
 JRST D,DIGIT;
 JSP BJA2;
 JRST BJA2;
 JRST BJA2;
 NOP; PERIOD;
 JRST TAB;
 JRST DASH;
 D,END; END OF STRING

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
ROUTINE TO APPROVE DEPARTMENT NAME OR NUMBER

PAGE 118

007763 200040 010577* 37510 BJA8: MOVE B,[POINT 7,F1 : OUTPUT POINTER
007764 201100 010014* 37520 MOVEI C,SCT3;
007765 402000 000005 37530 SETZM F
007766 403340 007721* 37540 SETZB H,SPARE2
007767 403400 000011 37550 SETZB I,J
007770 260740 010103* 37560 SCAN
007771 322400 010006* 37570 JUMP I,BJA9;
007772 404240 010600* 37580 AND F,[BYTE (7) 137,137,137]; FORCE UPPER CASE
007773 201040 000036 37590 MOVEI B,DEPTS;
007774 200101 010044* 37600 MOVE C,DEPT-1 (B)
007775 404100 010601* 37610 AND C,[BYTE (7) 177,177,177]; TAKE THREE CHARACTERS
007776 316100 000005 37620 CAMN
007777 254000 010002* 37630 JRST
010090 367040 007774* 37640 SOJG
010001 263740 000000 37650 B,BJA7+1;
DONE; FOUND
NO GOOD
010002 135040 010602* 37660 BJA8: LDB B,[POINT 7,DEPT-1 (B),35]; GET DEPT NUMBER
010003 202040 007766* 37670 MOVEM B,SPARE2;
010004 350017 000000 37680 AOS SAVE IN USER BLOCK
010005 263740 000000 37690 0 (PP);
DONE BUMP TO APPROVAL EXIT
010006 201040 000036 37700 BJA9: MOVEI B,DEPTS
010007 135100 010602* 37710 LDB C,[POINT 7,DEPT-1 (B),35]; GET DEPT NUMBER
010010 316100 000007 37720 CAMN C,H
010011 254000 010002* 37730 JRST BJA8; HE GOT ONE!
010012 367040 010007* 37740 SOJG AROUND FOR NEXT
010013 263740 000000 37750 B,BJA9+1;
DONE NONE FOUND - NOT OK

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
SCAN TABLE FOR DEPARTMENT APPROVAL

PAGE 119

010014	255000	000000	37790	SCT3:	NOP;	IGNORE BLANKS
010015	265140	010025*	37800	JSP	D,BJD1;	ALPHA
010016	265140	010032*	37810	JSP	D,BJD2;	NUMERIC
010017	265140	010025*	37820	JSP	D,BJD1;	SPECIAL - TREAT AS ALPHA
010020	254000	007622*	37830	JRST	BJA2;	ILLEGAL - DENY
010021	255000	000000	37840	NOP;		IGNORE PERIODS
010022	255000	000000	37850	NOP;		IGNORE TABS
010023	255000	000000	37860	NOP;		IGNORE DASH
010024	265140	007655*	37870	JSP	D,END;	END OF STRING
010025	326440	007622*	37880	BJD1:	JUMPN	ALPHA SCAN - DENY IF PRIOR NUMERIC
010026	350000	000010	37900	AOS	I	
010027	301400	000003	37910	CAIL	I,3;	NEED THREE ALPHAS
010030	254003	000001	37920	JRST	1(D);	ENOUGH SEEN ADD TO STRING AND EXIT
010031	254003	000003	37930	JRST	3(D);	ADD TO STRING
010032	326400	007622*	37940	BJD2:	JUMPN	NUMERIC SCAN - BAD IF ALPHA
010033	405100	000017	37960	ANDI	C,17;	MASK INTO NUMBER
010034	326440	010037*	37970	JUMPN	J,BJD22;	JUMP IF WE HAVE A DIGIT
010035	336000	000002	37980	SKIPN	C	
010036	254003	000000	37990	JRST	0(D);	IGNORE LEADING ZEROS
010037	221340	000012	38000	IMUL1	H,D10	
010040	270340	000002	38010	ADD	H,C;	COMPUTE BINARY
010041	350000	000011	38020	AOS	J	
010042	301440	000002	38030	CAIL	J,2;	NEED 2 DIGITS
010043	254003	000001	38040	JRST	1(D);	ENOUGH
010044	254003	000003	38050	JRST	3(D);	ADD TO STRING

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
DEPARTMENT NAME-NUMBER TABLE

PAGE 120

	RADIX	DEPT:	RADIX	DEPT:
010045	406111	500027	38070	38080
010046	406132	2000040	38090	38100
010047	406472	4000043	38110	38120
010050	416371	500121	38130	38140
010051	416372	300013	38150	38160
010052	416470	400121	38170	38180
010053	426310	500041	38190	38200
010054	426071	700012	38210	38220
010055	434270	100043	38230	38240
010056	434370	100043	38250	38260
010057	436131	700043	38270	38280
010060	462230	200025	38290	38300
010061	462370	700014	38310	38320
010062	466032	400120	38330	38340
010063	466152	300024	38350	38360
010064	464272	200032	38370	38380
010065	464372	200032	38390	38400
010066	502132	200034	38410	38420
010067	502213	100050	38430	38440
010070	512100	000074	38450	38460
010071	512132	000074	38470	38480
010072	512132	300036	38490	38500
010073	516130	300033	38510	38520
010074	516370	300132	38530	38540
010075	516470	400132	38550	38560
010076	516460	000132	38570	38580
010077	516631	700042	38590	38600
010100	516632	300042	38610	38620
010101	536032	300026	38630	38640
010102	536360	060026	38650	38660
			DEPTS=-DEPT	RADIX
				8

38410	;	C IS SCAN TABLE LOCATION, B IS OUTPUT BYTE POINTER LOCATION
38420	;	E LEFT IS CLOBBERED, SCAN TABLE RETURNS ARE: 0=IGNORE CHAR
38430	;	1=ADD TO OUTPUT STRING AND EXIT FROM SCAN, 2=EXIT FROM SCAN
38440	;	3=ADD CHARACTER TO OUTPUT STRING.
38450		
010103	271200 000003	SCAN: ADDI E,3; BUMP POINTER DOWN TO TEXT
010104	505200 440700	38460 E,4,40700; INPUT BYTE POINTER
010105	202200 002151*	HRLLI E,SCP1
010106	275200 000003	MOVM E,3; RESTORE BUFFER POINTER
010107	322040 010111*	SUBI E,3; SKIP IF CONTINUING OUTPUT
010110	202040 002153*	JUMPE B,*+2; OUTPUT BYTE POINTER
010111	542100 002155*	MOVEM B,SCP3; EXECUTE TABLE LOCATION
010112	134040 002151*	HRRM C,SC9;
010113	202040 000002	ILD B,SCP1; LOAD NEXT INPUT BYTE
010114	405040 000007	MOVEM B,C; SAVE THE BYTE
010115	211041 777770	ANDI B,7; MASK LOW ORDER
010116	240040 000002	MOVNI B,-8(B); COMPUTE 8-(B)
010117	137040 002154*	ASH B,2; 4 (8-(B))
010120	202100 000001	DPB B,SCP4; SET UP BYTE POINTER
010121	240040 777775	MOVEM C,B; RECOVER INPUT BYTE
010122	135040 002152*	ASH B,-3; HIGH 4 BITS FOR INDEX
010123	256020 002155*	LDB B,SCP2; GET THE TYPE
010124	254000 010112*	XCT aSC9; EXECUTE THE SCAN TABLE INSTRUCTION
010125	136100 002153*	JRST SC3; SKIP CHARACTER
010126	263740 000000	IDPB C,SCP3; ADD TO STRING AND EXIT
010127	136100 002153*	DONE; EXIT FROM SCAN
010130	254000 010112*	IDPB C,SCP3; ADD CHARACTER TO OUTPUT
		JRST SC3

38690 , TYPES ARE:
38700 , 0=BLANK
38710 , 1=ALPHA
38720 , 2=NUMERIC
38730 , 3=SPECIAL
38740 , 4=ILLEGAL
38750 , 5=PERIOD
38760 , 6=COMMA
38770 , 7=DASH
38780 , 8=END OF STRING
38790 ,
38800 , THERE ARE EIGHT 4 BIT CODES PER WORD WITH 4 ZEROS AT
38810 , THE END OF EACH WORD.
38820 ,
38830 ,
38840 , CART: BYTE (4) 8,4,4,4,4,4,4,0; COLUMN 1
38850 , BYTE (4) 3,6,3,3,8,8,4,0
38860 , BYTE (4) 4,4,4,4,4,4,4,0
38870 , BYTE (4) 4,4,4,4,4,4,4,0
38880 , BYTE (4) 0,3,3,3,3,3,3,0; COLUMN 2
38890 , BYTE (4) 3,3,3,3,3,3,7,5,3,0
38900 , BYTE (4) 2,2,2,2,2,2,2,2,0
38910 , BYTE (4) 2,2,3,3,3,3,3,0
38920 , BYTE (4) 3,1,1,1,1,1,1,0; COLUMN 3
38930 , BYTE (4) 1,1,1,1,1,1,1,0
38940 , BYTE (4) 1,1,1,1,1,1,1,0
38950 , BYTE (4) 1,1,1,3,3,3,8,0
38960 , BYTE (4) 4,1,1,1,1,1,1,0; COLUMN 4
38970 , BYTE (4) 1,1,1,1,1,1,1,0
38980 , BYTE (4) 1,1,1,1,1,1,1,0
38990 , BYTE (4) 1,1,1,4,4,4,4,0
010131 4 10421 042100
010132 154316 102100
010133 210421 042100
010134 210421 042100
010135 006314 631460
010136 146314 672460
010137 104210 421040
010140 104314 631460
010141 142104 210420
010142 042104 210420
010143 042104 210420
010144 042114 631600
010145 202104 210420
010146 042104 210420
010147 042104 210420
010150 042121 042100

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
JOSS LOG-OFF PROCESSOR PAGE 123

```

010151 254000 010152*   39010      EJ:          JRST      *+1
010152 201040 010160*   39020      MOVEI     B,EJ1
010153 202040 007470*   39030      MOVEM     B,INTENT;
010154 200040 007500*   39040      MOVE     B,PAGNO
010155 322040 010231*   39050      JUMPE    B,EJ3;
010156 201040 000001    39060      MOVEI     B,1;
010157 254000 004304*   39070      JRST      MONENT

010160 250740 000020*   39080      EXCH     PP,PPSAV
010161 260740 006421*   39100      TSX      ISEC
010162 202044 000002    39110      MOVEM     B,2 (E);
010163 315040 007507*   39120      CAMGE    B,ONTIME
010164 271040 250600    39130      ADDI     B,D3600*D24;
010165 274040 010163*   39140      SUB      B,ONTIME;
010166 264000 004262*   39150      JSR      COMPUTE SESSION TIME
010167 350002 002250*   39160      SESTIM(C) : INCREMENT PROPER COUNTER
010170 202044 000010    39170      MOVEM     B,8 (E);
010171 272040 002444*   39180      ADDM     B,T7+4
010172 200040 010603*   39190      MOVE     B,[XWD D14,1];
010173 202044 000001    39200      MOVEM     B,1 (E)
010174 200100 010546*   39210      MOVE     C,(POINT 7,3 (E))
010175 000500 000023*   39220      XMT     12,DATE
                                         OPDEF X [12B12]
                                         X DATE
                                         XMT 5,INITIALS
                                         OPDEF X [5B12]

010176 000240 007630*   X INITIALS      MOVE     B,SPARE1;
010177 200040 007665*   39240      MOVEM     B,6 (E)
010200 202044 000006    39250      MOVE     B,COMTIM
010201 200040 007477*   39260

```

```

010202 202044 000007    39280      MOVEM   B,7 (E) ;
                                DIVI   B,D10;
                                CVTL   JSR
010203 231040 000012    39290      AOS    CCTIM(C) ;
                                MOVE   S,CUI
010204 264000 004262*   39300      MOVE   SETZM
                                MOVEM  S,D9(E) ;
                                MOVE   B,USIZE
010205 350002 002262*   39310      MOVEM  B,D10(E)
                                MOVE   IMULI
                                LOW END = 10 CELLS
010206 200700 000021*   39320      JSR    CVTL
                                DSIZE(C) ;
                                LDB   B,S-BLOCK
010207 402016 002572*   39330      MOVEM  B,D11(E) ;
                                MOVE   S,-2;
                                MOVEI  IRREI
010210 202704 000011    39340      MOVE   B,PAGNO
                                MOVEM  B,D12(E) ;
                                MOVE   B,SPARE2
010211 200040 000000    39350      MOVEM  B,D13(E) ;
                                MOVE   B,SPARE3;
                                MOVEI  MONENT
010212 202044 000012    39360      JRST
010213 221040 000006    39370      COUNT IN SIZE DISTRIBUTION
010214 264000 004262*   39380      AOS    RECORD # OF CORE BLOCKS USED
010215 350002 002274*   39390      LDB   B,SPARE1
010216 135040 000264*   39400      MOVEM  B,D14(E) ;
                                MOVEI  GET CONTEXT FOR TAPE I/O
010217 202044 000013    39410      MOVEI  IRREI
010220 571700 777776    39420      MOVE   B,PAGNO
                                MOVEM  B,D12(E) ;
                                MOVE   B,SPARE2
010221 200040 010154*   39430      MOVE   B,D13(E) ;
                                MOVEI  CHARGE UNITS
010222 202044 000014    39440      MOVE   B,SPARE3;
                                MOVEI  MONENT
010223 200040 010003*   39450      JRST
010224 202044 000015    39460      MOVEI  TSX
                                MOVE   B,D14(E) ;
                                MOVEI  PUTB;
010225 200040 007503*   39470      EXCH   PP,PPSAV
                                MOVEI  B,D12;
                                MOVEI  TO MONITOR FOR FINAL OFF
010226 202044 000016    39480      MOVEI  MONENT
010227 260740 010305*   39490      JRST
010230 250740 000020*   39500      MOVEI  IRREI
010231 201040 000014    39510      EJ3:
010232 254000 004304*   39520      MOVEI  IRREI

```

		CHANGES STATE OF USER IN S TO STATE IN E (AN)	
		INDEX TO THE STATE TABLE	
		CLOBBERS B, C, E, AND F.	
		LDB	B,S,STA;
		CAMN	B,E
		DONE;	CURRENT STATE
		HRRZI	EXIT IF OLD STATE EQUALS NEW
		C,B;	ADDRESS OF HEADER
		MOVE	SAVE POINTER TO HEADER
		F,O(B);	CONTENTS OF ITEM
		JUMPN	WE HAVEN'T GOT SOMETHING WRONG, HAVE WE?
		F,*+2;	USER STATE AND QUEUE ARE DIFFERENT
		HALT	DOES ITEM POINT TO DESIRED USER?
		24;	YES
		CAIN	NEW ADDRESS OF ITEM
		CHS2;	
		JRST	
		B,O(F);	GET CELL TO BE UNLINKED
		HRRZI	TAKE OUT OF LIST
		CHS1	DID WE JUST MOVE UP THE LAST IN THE LIST?
		MOVE	NO
		F,O(F);	ADDRESS OF LAST ON LIST
		HRRM	END LINK TO HEADER
		F,O(B);	DOES END LINK POINT TO HEADER?
		TRUE	YEA VERILY--ZERO IT OUT
		F,-1;	UPDATE STATE
		CHS2;	
		JRST	
		CHS2,5;	
		MOVEI	
		HRLM	
		F,O(B);	
		CAMN	
		F,C;	
		GETZM	
		O(C);	
		DPB	
		E,S,STA;	
			PUT ON NEW LIST
		CHS4:	HRR,I
		SKIPN	E,S,QUE(E)
		JRST	B,O(E);
		CHS5;	BRING UP LIST HEADER AND
		SKIPN	JUMP IF LIST IS EMPTY
		JRST	SKIP IF LIST TOP SIGNAL IS ON
		CHS4,5;	NO--PUT ON BOTTOM OF LIST
		HRRM	NEW ITEM TO LIST TOP
		B,S,O(S);	RESTORE HEADER
		HRR,I	
		B,S,O(S);	
		MOVEM	
		B,O(E);	
		DONE	
			GET ADDRESS OF END OF LIST
			GET LAST ITEM ON LIST
			LINK IN CURRENT USER
			PUT ON END OF LIST
			END LINK TO LIST HEADER
			ZERO LAST IN LIST
		CHS4,5:	HLRZ
		B,O(E);	
		F,O(B);	
		HRR,I	
		F,S,O(S);	
		MOVEM	
		F,O(B);	
		HRLM	
		F,O(E);	
		HLLZS	
		S,O(S);	
		DONE	
		CHS5:	MOVEI
		HRL	F,S,O(S);
		F,F;	LINK AS TOP OF QUEUE
		MOVEM	F,O(E);
		HLLZS	S,O(S);
		DONE	ZERO END ITEM

```

40060 ; STATION # IN REGISTER S RIGHT
40070 ; INPUT IS POINTER TO BUFFER IN E RIGHT. ANYTHING IN LEFT
        ; B IS SCRATCH REGISTER. STARTS THE TRANSMISSION IF IT IS
        ; THE ONLY BUFFER ATTACHED.
40080 ;
40090 ;
40100 ;
40110 PUTB: CONO PI,1000+CHCTY; TURN OFF THE TRAPS
        ; CAME S,[DEC -2];
        ; JRST PUTB5;
        ; FSW B
        ; NOT TAPE
        ; QUIET TAPE SWITCH UP?
        ; YES - RETURN BUFFER
        ; PUTB4;
        ; JRST HRR2
        ; B,N.BUF B,5;
        ; FORGET TAPE OUTPUT IF FIVE OR
        ; FEWER BUFFERS AVAILABLE
        ; PUTB4;
        ; JRST HLRZ
        ; B,S.BUF (S) ;
        ; JUMPN B,PUTB1;
        ; JUMP IF BUFFERS ATTACHED
        ; ZERO LINK
        ; NEW BUFFER HEADER.
        ; TEST FOR JOSS STATION OUTPUT
        ; MOVEM E,S.BUF (S) ;
        ; PUTC2;
        ; S.PUTB2;
        ; JUMPGE S,[DEC -2];
        ; IS THE RANGE OK?
        ; HALT 25;
        ; XCT
        ; PUTB2 (S) ;
        ; JRST PUTB3
        ; TAPE I/O START
        ; CONSOLE TTY I/O START
        ; JOSS CONSOLE I/O START
        ; CHAIN NEW BUFFER TO LAST.
        ; ZERO CURRENT BUFFER LINK
        ; HLLZS O(E);
        ; E,S.BUF (S) ;
        ; PI,2000+CHDAT+CHCTY; TURN ON THE TRAPS
        ; DONE
        ; TSX MBA1;
        ; JRST PUTB3
        ; GIVE BACK THE BUFFER
        ; PUTB3

```

40420 ; EXITS +2 WITH BUFFER LOCATION IN E OR
40430 ; +1 IF NONE AVAILABLE.
40440 ; USES ONLY REGISTER E.

010343 700600 001002 GETBUF: CONO PI,1000+CHDAT+CHCTY; TURN OFF THE TRAPS
010344 371000 000401* SOSL COUNT DOWN AVAIL BUFFERS AND TEST
010345 254000 010350* N.BUF; JRST GET1
010346 402000 000401* SETZM N.BUF; NONE AVAILABLE. EXIT 1
010347 254000 010363* JRST GET2
010350 550200 000402* HRRZ E,L.BUF;
010351 540204 000000 HRR E,0 (E);
010352 504200 000402* HRL E,L.BUF;
010353 542200 000402* HRRM E,L.BUF;
010354 606200 777777 TRNN E,-1;
010355 402000 000402* SETZM L.BUF;
010356 554200 000004 40570 HLRZ E,E;
010357 513004 000000 40580 HLZS 0 (E);
010360 326200 010362* JMPN E,*+2
010361 040000 000011 HALT 11;
010362 350017 000000 AOS 0 (PP)
010363 700600 002002 GET2: CONO PI,2000+CHDAT+CHCTY; TURN ON THE TRAPS
010364 263740 000000 40630 DONE; TAKE EXIT 2

```

40650 ; RETURNS BUFFER ATTACHED TO S TO AVAILABLE.
40660 , CLOBBERS B,E
40670 MBA: CONO PI,1000+CHDAT+CHCTY; TURN OFF THE TRAPS
40680 HRRZ E,S.BUF(S); GET BUFFER HEADER
40690 SKIPN E
40700 HALT 1,2;
40710 H12: ZERO CANT BE A BUFFER
40720 HRRZ B,0 (E); GET LINK FROM ATTACHED BUFFER
40730 SKIPN B; SKIP IF NOT LAST ON LIST
40740 SETZM S,BUF(S); LAST, SO ZERO HEADER
40750 HRRM B,S.BUF(S); UPDATE HEADER
40755 HALT E,DUF; SKIP IF SPECIAL DISC BUFFER
40760 CALE E,DUF;
40770 JRST MBA1 FLAG OUTPUT COMPLETE
40780 AOS SKT; AND DONT PUT ON AVAILABLE
40790 JRST MBA5;

40800 ; RETURN BUFFER IN E TO AVAILABLE
40810
40820 MBA1: CONO PI,1000+CHDAT+CHCTY; TURN OFF THE TRAPS
40830 HRRZ B,L.BUF THIS BUFFER SHOULD NOT BE ON THE LIST ALREADY
40840 CAMN B,F; BUT IT IS!!!
40850 HALT 1,3;
40860 JUMPE B,,+3; IF WE ARE AT THE END
40870 HRRZ B,O(B); GET THE NEXT ON THE LIST
40880 JRST MBA4; AND GO TO TEST IT
40890 HALT B,L.BUF; ADDRESS OF LAST ON AVAILABLE
40900 JUMPE B,MBA3; ARE THERE NONE? JUMP IF SO.
40910 HRRM E,O(B); LINK OBJECT BUFFER TO LAST
40920 HRLM E,L.BUF; AND TO HEADER
40930 SETZM O(E); ZERO HEADER OF OBJECT BUFFER
40940 AOS N.EUF; COUNT ONE MORE AVAILABLE
40950 CONO PI,2000+CHDAT+CHCTY; TURN ON THE TRAPS
40960 DONE

40980 MBA3: HRRM E,L.BUF; SET UP FOR ONLY BUFFER ON LIST
40990 JRST MBA2
41000
010420 542200 000402*
010421 254000 010413*

```

41020 ; ROUTINE AT CALLER + 1 IS EXECUTED TO FILL BUFFER
41030 ; CLOBBERS S,E,C,B.
010422 260740 010343*
010423 254000 010431*
010424 571700 777777
010425 200100 010543*
010426 256037 000000
010427 000140 004134*
010430 260740 010305*
010431 350017 000000
010432 263740 000000
41040 CMSS: TSX GETBUF
41050 CM1
41060 JRST
41070 HRREI S,-1; CONSOLE OUTPUT CONTEXT
41080 MOVE C,[POINT 7,1 (E)]; POINT TO BUFFER
41090 XCT A0 (PP); DO ROUTINE TO FILL THE BUFFER
41100 XMT 3,CRLF; TERMINATE THE BUFFER
OPDEF X [3B12]
X CRLF
010430 PUTB; TRANSMIT IT
010431 AOS 0 (PP); BUMP TO EXIT
010432 DONE
41140

41160					
41170					
41180					
41190					
010433	554040	000004	41200	B,E;	ADDR OF BLT CONTROL WORD IN MESSAGE
010434	200100	000004	41210	C,E;	SAVE BUFFER POINTER
010435	200041	000000	41220	B,O (B);	PICK UP XWD FIRST LOC. CT
010436	500200	000001	41230	HLL E,B;	FIRST LOC TO BLT COUNT WORD
010437	271200	000003	41240	ADDI E,3;	BUMP TP TEXT LOCATION
010440	270040	000004	41250	ADD B,E;	CT + FIRST LOC = LAST WD + 1
010441	251201	777777	41260	BLT E,-1 (B);	TRANSFER MSG TO BFR
010442	200200	000002	41270	MOVE E,C;	RESTORE BUFFER POINTER
010443	263740	000000	41280	DONE	

AUG 16 1967
1000

010444 010445•000011 41300 XWD *+1, BJM1E-; JOSS AT YOUR SERVICE. INITIALS PLEASE:
010445 066251 751646 41310 ; BYTE (7) 15, 112, 117, 123, 40, 141, 164, 40, 171;
010446 203036 420362 41320 BYTE (7) 157, 165, 162, 40, 163, 145, 162, 166, 151, 143;
010447 677536 220346 41330 BYTE (7) 145, 56, 15, 111, 156, 151, 164, 151, 141, 154;
010450 627456 664706 010451 625341 544734 41340 BYTE (7) 163, 40, 160, 154, 145, 141, 163, 145, 72, 40;
010452 647515 160730 010453 715016 066312 41350 BYTE (7) 145, 56, 15, 111, 156, 151, 164, 151, 141, 154;
010454 607474 535100 010455 000000 000000 41360 BYTE (7) 0;
010456 010457•000004 41370 XWD *+1, BJM2E-; PROJECT NUMBER PLEASE:
010457 503455 765312 41380 BYTE (7) 120, 162, 157, 152, 145, 143, 164, 40, 156, 165
010460 617504 067352 010461 667054 571164 41390 BYTE (7) 155, 142, 145, 162, 72
010462 200000 000000 41400 BYTE (7) 40, 0
010463 010464•000006 41410 XWD *+1, BJM3E-; ONE TO FOUR LETTERS PLEASE:
010464 477354 520350 41420 BYTE (7) 117, 156, 145, 40, 164, 157, 40, 146, 157, 165
010465 675014 667752 010466 711015 462750 41440 BYTE (7) 162, 40, 154, 145, 164, 145, 162, 163, 40
010467 723136 271500 010470 703314 560746 41450 BYTE (7) 160, 154, 145, 141, 163
010471 625644 000000 41460 BYTE (7) 145, 72, 40, 0
010472 010473•000007 41470 XWD *+1, BJM4E-; PLEASE USE A LEGITIMATE NUMBER:
010473 503314 560746 41480 BYTE (7) 120, 154, 145, 141, 163, 145, 40, 165, 163, 145
010474 625016 571712 010475 203024 066312 41500 BYTE (7) 40, 141, 40, 154, 145, 147, 151, 155
010476 637236 464732 010477 607514 520334 41510 BYTE (7) 141, 164, 145, 40, 156, 165, 155, 142, 145, 162
010500 727334 262744 010501 351000 000000 41520 BYTME: BYTE (7) 72, 40, 0
010502 010503•000007 41530 XWD *+1, BJM5E-; IF YOU ARE LEARNING USE "1" :
010503 447144 074736 41540 BYTE (7) 111, 146, 40, 171, 157, 165, 12, 40, 141, 162
010504 724244 060744 010505 625015 462702 41550 BYTE (7) 145, 40, 154, 145, 141, 162, 156, 151, 156, 147
010506 713355 167316 010507 203536 362500 41560 BYTE (7) 40, 165, 163, 145, 40, 42, 61, 42, 40, 72
010510 211424 220164 010511 200000 000000 41570 BYTE (7) 40, 0

JOSS SUPERVISOR 8/1/67 COPYRIGHT 1966 THE RAND CORP.
LOG-ON AND SHUTDOWN MESSAGES

PAGE 132

010512	010513•000005	41600	SDM:	XWD	*+1, BJM6E-;	SYSTEM SHUTTING DOWN
010513	066477 171750	41610	BYTE	BYTE	(7) 15, 123, 171, 163, 164, 145, 155, 40, 163, 150	
010514	627324 071720					
010515	727516 464734	41620	BYTE	(7)	165, 164, 164, 151, 156, 147, 40, 144, 157, 167	
010516	635014 467756					
010517	671341 500000	41630	BJM6E:	BYTE	(7) 156, 56, 15, 0	
010520	010521•000003	41640	BJM7:	XWD	*+1, BJM7E-;	DEPARTMENT:
010521	423136 060744	41650	BYTE	BYTE	(7) 104, 145, 160, 141, 162, 164, 155, 145, 156, 164	
010522	723334 567350					
010523	351000 000009	41660	BJM7E:	BYTE	(7) 72, 40	
010524	010525•000006	41670	BJM8:	XWD	*+1, BJM8E-;	DEPARTMENT NAME OR NUMBER:
010525	423136 060744	41680	BYTE	BYTE	(7) 104, 145, 160, 141, 162, 164, 155, 145, 156, 164	
010526	723334 567350					
010527	203354 166712	41690	BYTE	(7) 40, 156, 141, 155, 145, 40, 157, 162, 40, 156		
010530	203376 220334					
010531	727334 262744	41700	BYTE	(7) 165, 155, 142, 145, 162		
010532	351000 000000	41710	BJM8E:	BYTE	(7) 72, 40, 0	

010533 720100 003527* 41750
010534 264000 002745*
010535 440700 000000
010536 440700 004563*
010537 360300 000040
010540 270700 000040
010541 000026*000031
010542 777760 000120*
010543 440704 000001
010544 440600 002216*
010545 476150 626400
010546 440704 000003
010547 000011 000011
010550 777777 777776
010551 000272*000272*
010552 000020 000020
010553 100000 700000
010554 440500 007333*
010555 440500 005325*
010556 260740 005313*
010557 440500 005324*
010560 440500 005322*
010561 260740 005365*
010562 000000 003305*
010563 440500 005542*
010564 440716 002572*
010565 350700 006377*
010566 440700 006373*
010567 777730 000000
010570 360507 002745*
010571 221407 002745*
010572 440500 007236*
010573 440700 000023*
010574 440700 000025*
010575 375767 720100
010576 202371 626400
010577 440700 000005
010600 576773 700000
010601 777777 700000
010602 000701 010044*
010603 000016 000001

THERE ARE NO ERRORS

PROGRAM BREAK IS 010604

SYMBOL TABLE

A 000000
ABG 002134*
ABG-S 000017
ACTION 006025* EXT
ADATE 007212* INT
ADIS 004650*
ALPH1 007624*
ALPHA 007652*
APRR 003766* INT
ATIM 000025*
B 000001
BBLOCK 000020
BEEPS 400000
BELF 004135*
BIO 003602*
BJ 007462*
BJ1 007514*
BJ2 007517*
BJ2.1 007524*
BJ3 007526*
BJ3.1 007531*
BJ3.2 007536*
BJ3.3 007534*
BJ4 007541*
BJ5 007562*
BJ6 007567*
BJ7 007572*
BJ8 007576*
BJ9 007601*
BJ9.1 010433*
BJA1 007605*
BJA2 007622*
BJA3 007626*
BJA4 007627*
BJA5 007656*
BJA5.2 007672*
BJA7 007773*
BJA8 010002*
BJA9 010006*
BJAD 007763*
BJAN 007723*
BJAY 007722*
BJD1 010025*
BJD2 010032*
BJD22 010037*
BJLEV 007546*
BJM1 010444*
BJM1E 010455*
BJM2 010456*
BJM2E 010462*
BJM3 010463*
BJM3E 010471*
BJM4 010472*
BJM4E 010501*
BJM5 010502*

BJM5E	010511*
BJM6E	010517*
BJM7	010520*
BJM7E	010523*
BJM8	010524*
BJM8E	010532*
BJS1	007702*
BJS2	007711*
BJS3	007713*
BJS4	007720*
BJSU	007745*
BJSW	007753*
BLANKS	004130*
BLKSWR	003332*
BLTC	005073*
BOF1	005521*
BOF10	005542*
BOF2	005525*
BOF3	005533*
BOF5	005536*
BOFF	005512*
BTA	004715*
BTA1	004722*
BU2	004437*
BUF	000403*
BUF1	004442*
BUFBIT	000004
BUFFER	004430*
BUFSIZ	000022
C	000002
C20	000000 EXT
C27	010332 EXT
C28	006602 EXT
C3	004136*
C30	004024* EXT
C31	006266* EXT
C32	004072* EXT
C4	004137*
C5	004140*
CAPR	000007
CART	010131*
CCTIM	002262*
CCTY	000006
CDAT	000001
CDC	000000
CDRM	000002
CH630	000030
CHAPR	000001
CHCTY	000002
CHDAT	000100
CHDC	000040
CHDRM	004141*
CHECK	00451*
CHECK1	004176*

CHK1	004174*
CHK2	004206*
CHS*	010233*
CHS1	010240*
CHS2	010247*
CHS2.5	010257*
CHS4	010260*
CHS4.5	010271*
CHS5	010300*
CI630	000003
CIO	003475*
CIO1	003524*
CK	002130*
CK.S	000013
CKER	003162*
CKF	002114*
CKS10	003232*
CKS11	003233*
CKS5	003225*
CKS7	003231*
CKSUM	003207*
CL1	004055*
CL10	000036*
CL11	000037*
CL12	000040*
CL13	000041*
CL14	004075*
CL15	004104*
CL16	004120*
CL17	004125*
CL18	004014*
CL2	004057*
CL3	004051*
CL4	004025*
CL5	004015*
CL6	004063*
CL7	004065*
CL8	004010*
CL9	000035*
CLOCK	003771*
CM1	010431*
CMESS	010422* INT
CO.BUF	000330*
CO630	000004
COM	002124*
COM.S	000007
COMEBA	002111* INT
COMP1	005133*
COMP2	005140
COMPAC	005122*
COMTIN	010201* EXT
CONSOL	005546* EXT
CONT	004355*
CONT1	004361*
CORBIT	000020

CORE 000272, INT
CPIL 002332,
CPOL 002344,
CR,R 006457,
CRLF 004134,
CRS 004133,
CSS 002113,
CT10 002156,
CT10A 002157,
CT11 002204, INT
CT12 002200,
CT13 002165, INT
CT14 002166, INT
CT15 002167, INT
CT16 002171, INT
CT17 002210, INT
CT18 002204, INT
CT18A 002205, INT
CT18B 002206, INT
CT19 002207, INT
CT19A 002203,
CT2 002162,
CT21 002224,
CT22 002226, INT
CT22A 002225, INT
CT23 002211, INT
CT24 002227, INT
CT25 002174,
CT26 002175,
CT27 002170,
CT27A 002176, INT
CT27B 002177, INT
CT28 002164, INT
CT2A 002174,
CT31 002173, INT
CT32 002160,
CT33 002161,
CT34 002162,
CT35 002163,
CT37 002172, INT
CT39 002202,
CT3A 002214,
CT41 002221,
CT42 002222,
CT43 002212,
CT44 002213,
CT45 002223,
CT46 002214, INT
CT47 002215, INT
CT48A 002216,
CT48B 002217,
CT48C 002220,
CT49 000320,
CT50 002230,
CT7 002221,

CT7A	002230*
CTD	006403* INT
CTD1	006414*
CTD2	006416*
CTDB	006405* INT
CTDR	006410* INT
CTPAR	002712*
CTPARS	000033
CTYR	003544* INT
CU	002125*
CU.S	000010
CUI	000021*
CVT1	006511*
CVTL	004262*
CVTL1	004267*
CVTLIN	006507*
CVTLOG	004251*
CVTS	000011
D	000003
D.TIME	003772* EXT
DAM	000001
DASH	007653*
DATE	000023* INT
DATETIME	000000 EXT
DAY	000030* INT
DBASE	000321*
DBUF	010375* EXT
DCLOB	040000
DCLOBM	004577*
DCOMP	004625*
DCONT	004616*
DCONT1	004622*
DCP1	004635*
DCPI	002320*
DCT	002126*
DCT.S	000011
DDT	000000 EXT
DE10	003154*
DECT	003163*
DEDR	003164*
DEDR1	003165*
DEER	003126*
DELTA	000004
DEPT	010045*
DEPTS	000036
DERR	003133*
DI1	006646*
DI2	006653*
DI3	006661*
DI4	006663*
DIG1	007647*
DIGIT	007643*
DIP	002127*
DIP.S	000012
DISC	000000 EXT

DISC-C	006026*	EXT
DISC-D	004667*	EXT
DISC-S	007276*	EXT
DISC1	005610*	
DISC2	005600*	
DISCP	005573*	
DISINT	006645*	
DISMIS	002745*	
DKBIT	000010	
DKBY	002110*	
DM10	003113*	
DM90	007322*	
DMBY	002107*	
DMERR	003137*	
DMIBK	003324*	
DMIN	003331*	
DMNR	003305*	
DMT	000020	
DMUSR	003327*	
DMWR	003330*	
DOAF	000010	
DOF	006664*	
DOF1	006673*	
DP	000010	
DPWD	003325*	
DPWD1	003326*	
DQ	002131*	
DQ-S	000014	
DR	000400	
DR2	003014*	
DREQ	004604*	
DREQ1	004613*	
DRM10	003104*	
DRM11	003105*	
DRM12	003106*	
DRM13	003107*	
DRM14	003110*	
DRM2	003037*	
DRM3	002762*	
DRM4..5	003031*	
DRM5	003032*	
DRM6	003033*	
DRM6..5	003035*	
DRM7	003043*	
DRM7..5	003061*	
DRM7..6	003077*	
DRM7..7	003100*	
DRM8	003102*	
DRM9	003103*	
DRMPP	003111*	
DRMR	002750*	INT
DSIZE	002274*	
DSS	006101*	
DSTAT	000020	
DSTRT	004642*	

DSU	DSU.S	002133*
	DT.BUF	000016
	DTIM	000327*
	DUI	002306*
	DUMP	000002
E		0000000 EXT
		000004
EJ		010151*
EJ1		010160*
EJ2		010161*
EJ3		010231*
END		007655*
END.S		000037
ENTSW		004327*
F		000005
F1		005476*
F2		005500*
F3		005507*
FAIL		003166*
FAKE		006171*, EXT
FILE		004656*, EXT
FIND		005472*
FINDB		005032*
FM		005152*
FM1		005153*
FM2		005161*
FM3		005163*
FM4		005173*
FM5		005174*
FM7		005203*
FRMT		007240*
G		000006
GET1		010350*
GET2		010363*
GETBUF		010343*
GETCT		006612*
GRN		000247*
GR		002132*
GR,S		000015
GR1		005614*
GR2		005616*
GR3		005621*
GRONK		005613*
GT1		006630*
GTI		005624*
H		000007
H1		007414*
H10		005603*
H11		010361*
H12		010370*
H13		010404*
H14		003235*
H15		002755*
H17		002777*
H2		004177*

H20	006224*
H21	006056*
H22	005453*
H23	004737*
H24	010242*
H25	010325*
H26	004215*
H27	004145*
H3	003713*
H30	007343*
H31	004156*
H33	004124*
H5	004325*
H6	006142*
H7	006520*
HALT\$	020000
HDM	004535*
HEAD	002231*
HMES	007100* INT
HMES1	007103*
HR	000010
HSDM	004571* INT
I	000010
IINT	002356*
IN10	006427*
IN11	006432*
IN20	006434*
IN30	006436*
IN40	006441*
IN50	006454*
IN99	006452*
INBIT	000002
INIT IA	010176* EXT
INSR	003727*
INT1	007446*
INT2	007423*
INT3	007431*
INTBEG	007566* EXT
INTENT	010153* EXT
INTINT	007406*
IRPTR	004127*
ISEC	006421*
ISWAP	003333*
J	000011
JOBNO	007657* EXT
K	000012
K1	000074
K2	000054
K3	000017
KEY	004660* EXT
KILL	004162* INT
KILL1	007456*
L	000013
LBUF	000402*
LFTSW	000021

L. OPM	004562*
LASTB	002064*
M	000014
M1	005010*
M2	005100*
M25	005111*
M4	005116*
M5	005135*
M51	005143*
M52	005157*
MBA	010365*
MBA1	010401*
MBA2	010413*
MBA3	010420*
MBA4	010403*
MBA5	010416*
MDS	006007*
MDS2	006030*
MIN	000032* INT
MINT	002572*
MISC	002112*
MONE1	004351*
MONE NT	004304* INT
MONEXI	004350*
MONS	007236*
MONTH	000027* INT
MOR1	005050*
MOR2	005033*
MOR2.5	005034*
MOR3	005041*
MOR3.5	005046*
MOR4	005052*
MOR41	005056*
MOR5	005064*
MOR51	005070*
MOR6	005020*
MOR7	005025*
MORC1	004764*
MORC1	005013*
MORCOR	004760*
MSCAN	010107*
MSGPR	005543*
MSGPR1	005555*
MT1	00224
MT2	00230
MTC	00220
N	000015
N. BUF	000401*
N.C	000313*
N.CB	000312* INT
N.CK	000323*
N.COR	000020
N.DRM	000320*
N.PP1	000020
N.S	000050

643 5482.16 15

N.SG	000050
N.SON	000324* INT
N.UC	000322*
NEXT	007550*
OCT	006700*
OCT1	006703*
OCTW	004274* INT
OCTW1	004276*
ODIS	006711*
OF	002140*
OF.S	000023
OF1	006302*
OF2	006304*
OFF	006276*
OFF5	006321*
OFFD	006323*
OFFNQ	006274*
OFFQ	006261*
OFFS	100000
OGKM	004211*
OINT	002370*
OLSW	000100
ON	002116*
ON.R	006326*
ON.S	000001
ON5	006337*
ONBIT	000040
ONTIME	010165* EXT
OPMSG	004563*
OSWAP	003234*
PAGE	004501*
PAGNO	010221* EXT
PATCH	000000*
PAUSE	004416*
PDEP	007735*
PG1	004557*
PG2	004550*
PG3	004544*
PG4	004551*
PHIG	007740*
PLOW	007737*
PP	000017
PP1	000121*
PPSAV	000020*
PPW	000120* INT
POP	005561*
POP2	005571*
PR1	003706*
PRMES	000040
PROG	004672* EXT
PROP	003673* INT
PRSIG	006134*
PSTN	007736*
PUTB	010305*
PUTB1	010334*

PUTB2	010332*
PUTB3	010337*
PUTB4	010341*
PUTB5	010316*
Q2	004465*
QC	002123*
QC.S	000006*
QCTR	006604*
QDN	002136*
QDM.S	000021*
QM	002137*
QM.S	000022*
QM1	006371*
QM2	006377*
QM3	006403*
QM4	006373*
QP	002135*
QP.S	000020*
QUIT	004446*
R2	003732*
R3	003735*
R4	003740*
R4.1	003750*
R4.99	003747*
RC	002117*
RC.S	000002
RCBIT	000001
RCOR	004726*
REBUF	004444*
RESK	006013*
RESULT	006033* EXT
RI	002120*
RI.S	000003
RIB	002121*
RIB.S	000004
RISIG	007434* EXT
RJD	003765* EXT
RPN	007741*
RPNS	000011
RSIG	005636* INT
RUL	005050*
RUL1	005052*
RUL2	005067*
S	000016
S.BLOC	000064*
S.BUF	000331* INT
S.COR	000263*
S.DU	000267*
S.GK	000270*
S.ID	000315*
S.INR	000266*
S.IU	000314*
S.M	000002
S.OFR	000265*
S.OK	000325* INT

S.Q	000141*
S.QUE	002115* INT
S.S	000211* INT
S.SIG	000271*
S.STA	000261*
S.TM	000262*
S.UD	000316*
S.UR	000317*
S10	005322*
S11	005324*
S12	005325*
SA	000112
SA1	006250*
SAQM	006246*
SC3	010112*
SC9	002155*
SCAN	010103*
SCP1	002151*
SCP2	002152*
SCP3	002153*
SCP4	002154*
SCT1	007632*
SCT2	007752*
SCT3	010014*
SDM	010512*
SDP	006104*
SDS	200090
SEC	000033* INT
SECOND	000044* INT
SELINT	007335*
SELSWP	005211*
SEQ	007603* EXT
SESTIM	002250*
SG	006560*
SG.L	000115* INT
SG.LIM	000117* INT
SG.M	000116* INT
SHUT	006126* EXT
SI1	007333*
SI2	007337*
SI3	007346*
SI3.5	007355*
SI4	007360*
SI4.5	007362*
SI7	007367*
SI8	007376*
SIGPR	007264* INT
SIGPR1	007264*
SIGTBL	000045* INT
SK1	006055*
SK10	006075*
SK11	006077*
SK2	006070*
SK3	006041*
SK4	006061*

SK5	006072*
SKR	006103*
SKT	006102*
SKULK	006032*
SOUT	007045* INT
SP1	006135*
SP2	006154*
SP3	006146*
SPARE1	010177* EXT
SPARE2	010223* EXT
SPARE3	010225* EXT
SPARE4	007441* EXT
SQM	006342*
SRPN	007724*
SRPNS	000011
SRR	005110*
SS1	005216*
SS10	005262*
SS11	005274*
SS15	005277*
SS17	005305*
SS18	005313*
SS2	005224*
SS3	005235*
SS30	005327*
SS32	005335*
SS33	005351*
SS34	005353*
SS35	005365*
SS4	005240*
SS40	005376*
SS41	005410*
SS42	005417*
SS44	005437*
SS45	005442*
SS5	005242*
SS6	005246*
SS90	005456*
SS98	005470*
SS99	005471*
SSIG	005650* INT
ST.1	007063*
ST.12	007077*
ST1	007114*
ST1A	007136*
ST1B	007115*
ST2	007141*
ST2.4	007157*
ST2.5	007131*
ST2.6	007135*
ST22	003574*
ST23	003575*
ST24	003576*
ST3	007243*
ST31	003577*

ST32	003600*
ST33	003557*
ST34	003555*
ST35	003601*
ST4	007246*
ST40	003572*
ST41	003612*
ST42	003627*
ST43	003643*
ST44	003652*
ST45	003660*
ST46	003664*
ST47	003665*
ST48	003666*
ST49	003667*
ST5	007254*
ST50	003670*
ST51	003671*
ST52	003672*
ST6	007256*
ST7	007263*
STA1	006775*
STA1.5	007025*
STA2	007043*
STA3	006762*
STAT	006737*
STAT1	006716*
STAT2	000020
SU	004372*
SU.5	004401*
SU1	004404*
SU2	004406*
SU3	004414*
SUM	002642*
SWITCH	010310* EXT
T.CU	000043*
T.MAX	000326*
T1	006200*
T10	002476* INT
T10.	002141*
T11	002510* INT
T11.1	002511* INT
T11.2	002512* INT
T11.3	002513* INT
T11.4	002514* INT
T11.5	002515* INT
T11.6	002516* INT
T11.7	002517* INT
T11.8	002520* INT
T11.9	002521* INT
T2	006173*
T3	006224*
T5	002414*
T6	002426*
T7	002440*

SYMBOL TABLE

T7.9	002451*	INT
T8	002452*	INT
T80	007244*	EXT
T80..99	007250*	EXT
T9	002464*	INT
TABO	007057*	
TABO1	007062*	
TAPES	003526*	
TCW	003527*	
TE1	005446*	
TE2	005454*	
TE2.0	003432*	
TE21	003434*	
TE22	003436*	
TEND	003410*	
TEND1	003424*	
TEND2	003425*	
TEND4	003444*	
TEND5	003445*	
TEND6	003446*	
TEND7	003447*	
TEND8	003450*	
TEND9	003451*	
TEND93	003452*	
TEND95	003461*	
TERASE	002222*	INT
TERM	006130*	
TERROR	002221*	
TEST	005445*	
TICK	000034*	
TIME	000042*	INT
TINT	002402*	
TIP	003405*	
TL	004366*	
TLSU	004370*	
TLT	004704*	
TLT1	004706*	
TO	006517*	
TO1	006536*	
TO3	006553*	
TO99	006555*	
TOCK	006542*	
TOCK1	006552*	
TODSU	006537*	
TOF	002115*	
TOF.S	000000	
TPDL	003530*	
TPQ	000004	
TRST	003532*	
TRST1	003541*	
TSIG	005643*	INT
TTT	005667*	
TWATT	003470*	
TYPE6	000022*	INT
U	000001	

UC	UC.S	0002122.
UD2		0000005
UPDATE		0007176.
UP1		0007162.
USERS		0007212.
USIZE		0002160. INT
USTAT		010211. EXT
V		002522.
W		0000002
WAIT		0000003
WJD		004420.
X		0000000
XMTR		000004
Y		003716.
YEAR		0000005
Z		0000026. INT
ZERO		0000006
ZZ		004132.
		005050.

END OF ASSEMBLY

600 5051

165

-153-

PART II

PREAMBLE AND RECOVERY

00040 ; JOSS INITIALIZATION AND RECOVERY PROGRAMS
00050
00060
00070 ; G. E. BRYAN
00080
00090
00100 ; ASSEMBLED 8/1/67 FROM TAPE 56 (A SEQUENCED COPY OF 31)
00110 ; BINARIES ON TAPE 24 AS 10

```

00130 ; INITIALIZATION DIALOGUE FOR JOSS STARTUP.
00140      ; THIS CODE MUST BE LOADED LAST, DIRECTLY
00150      ; PRECEDED BY DDT; A (THE ARITHMETIC) MUST
00160      ; BE LOADED FIRST TO PRESERVE A DDT ENTRY AT 140.
00170      ;
00180      ; ASSEMBLY PARAMETERS FOR THE PI CHANNELS
00200
00210      CDC=1; 1 - DATA CONTROL
00220      CDRM=2; 2 - I/O PROCESSOR AND DRUM CONTROL
00230      CI630=3; 3 - 630 INPUT
00240      CO630=4; 4 - 630 OUTPUT
00250      CDAT=6; INTERRUPT CHANNEL FOR TAPE
00260      CCTY=6; 6 - CONSOLE TELETYPE
00270      CAPR=7; 7 - PROCESSOR
00280
00290      CHDC=100; THESE ARE CORRESPONDING DEFINITIONS FOR PI CONOS
00300      CHDRM=40;
00310      CH630=30
00320      CHDAT=0
00330      CHCTY=2
00340      CHAPR=1
00350      MTC=220;
00360      MT=224
00370      MT2=230
00380      DR=400
00390      DP=010
00400      DMT=20;
00410      DDR=50;
00420
00430      A=0
00440      B=1
00450      C=2
00460      D=3
00470      E=4
00480      F=5
00490      G=6
00500      H=7
00510      I=10
00520      J=11
00530      K=12;
00540      L=13;
00550      M=14;
00560      N=15;
00570      S=16;
00580      PP=17;
00590
00600      DEFINE XMT (M,A); TRANSMIT M CHARACTERS FROM A
00610      <OPDEF X [M,B12]
00620      X A>
00630
00640      OPDEF TSX {PUSHJ PP,0}
00650      OPDEF DONE {POPJ PP,0}

```

INTERNS AND EXTERNS

00670 EXTERN DDT,MONTH,YEAR,HR,MIN,ADATE,DATE
00680 EXTERN CMESS,PROP,DRMR,C25,C26,CTYR,APRR
00690 EXTERN OCTW
00700 EXTERN CORE,C20,PPW,TYPE6,SOUT,HMES,SIGPR
00710 INTERN DUMP,RJD,WJD,DATIME
00720 INTERN SWITCH,FAKE,L100,BEGIN
00730
00740
00750
00760
00770 DISMIS: Z
000000 000000 000000 CONO PI,1000+CHDC;
000001 700600 001100 00780 12,DISMIS; TURN OFF INTERRUPT CHANNEL
000002 254520 000000 00790 JRST DISMISS INTERRUPT

L02 50000 163

000003	700600	000200	00820	SAVE=120;	SAVE LOCATION FOR THE ACS
	DUMP:	I OA=777700	CONO	PI,200;	TURN ON THE INTERRUPT SYSTEM
000004	250040	000564•	00830	EXCH	B,[XWD 0,SAVE]
000005	251040	000137	00840	BLT	B,SAVE+17;
000006	250040	000564•	00860	EXCH	B,[XND 0,SAVE]
000007	202040	000121	00870	MOVEM	B,SAVE+1
000010	722200	000000	00880	CONO	MTC,0
000011	722600	000002	00890	CONO	MT1,2;
000012	200040	000565•	00900	MOVE	B,[JSR DISMISS]
000013	202040	000043	00910	MOVEM	B,4142*CDC
000014	200040	000566•	00920	MOVE	B,[BLKO DC,S]
000015	202040	000042	00930	MOVEM	B,4042*CDC
000016	201200	000000	00940	MOVEI	E,0;
000017	201240	000020	00950	MOVEI	F,20;
000020	200140	000567•	00960	MOVE	D,[POINT 3,SAVE]
000021	254000	000024•	00970	JRST	TD1;
000022	306200	000020	00980	TD05:	CAIN
000023	200140	000570•	00990	MOVE	E,20;
000024	200300	000571•	01000	MOVE	D,[POINT 3,20];
000025	201040	000000	01010	MOVE	G,[POINT 6,10];
000026	136240	000006	01020	MOVEI	B,0
000027	200100	000572•	01030	IDPB	F,G;
000030	134000	000002	01040	MOVE	C,[POINT 3,E,D20];
000031	136000	000006	01050	ILDB	CARRIAGE CONTROL
000032	305040	000004	01060	A,C	A POINT TO LOCATION COUNTER
000033	344040	000030•	01070	IDPB	A,G;
000034	201040	000010	01080	CAIGE	ADDRESS
000035	136240	000006	01090	B,TD2	
000036	136240	000006	01100	MOVEI	
000037	201340	000014	01110	TD3:	B,10;
000040	134000	000003	01120	IDPB	F,G;
000041	136000	000006	01130	IDPB	H,14;
000042	367340	000040•	01140	MOVEI	A,D
000043	367040	000035•	01150	IDPB	TWO BLANKS BETWEEN WORDS
000044	136240	000006	01160	SOJG	H,14;
000045	136240	000006	01170	IDPB	A,D
000046	722740	040000	01180	CONSO	CONVERT TO BCD
000047	254000	000046•	01190	JRST	H,-2
000050	200700	000573•	01200	MOVE	SOJG
000051	720200	003421	01210	CONO	JUMP IF LINE NOT FULL
000052	722200	031000	01220	IDPB	I OA=40000;
000053	700600	002100	01230	CONO	WAIT FOR XFER NEW COMMAND
000054	700700	000100	01240	CONO	S,[XND -D20,I OA-1];
000055	254000	000054•	01250	CONSZ	WAIT FOR TAPE CONTROL FREE
000056	271200	000010	01260	JRST	DC,CDC*DMT+3400
000057	305200	100000	01270	ADDI	PI,2000+CDC
000060	254000	000022•	01280	CAIGE	PI,CHDC;
000061	722740	040000	01290	JRST	WAIT FOR DATA CONTROL FINISHED
000062	254000	000061•	01300	CONSO	-1;
000063	722200	001400	01310	JRST	E,D8
000064	722740	040000	01320	CONSO	E,100000;
000065	254000	000064•	01330	JRST	MT1,400000;
000066	722200	000400	01340	CONO	-1;
000067	254000	000000	01350	JRST	MTC,4000;
					END FILE
					WAIT FOR COMMAND OK
					REWIND
					GO TO DDT

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 5
ROUTINE TO INPUT DATE AND TIME

```

000070 712200 003600    01370   DATIME: CONO      TTY,3600;
000071 260740 000423*   01380   TSX      SLM;
000072 260740 000000    01390   TSX      ALATE;
000073 260740 000172*   01400   TSX      D50;
000074 260740 000254*   01410   TSX      DTO;
000075 403140 000007    01420   SETZB  D,H;      SLASH COUNT AND CONVERTER
000076 260740 000274*   01440   TSX  LM;
000077 201040 000006    01450   MOVEI  B,6;
000100 200100 000574*   01460   MOVE  C,[POINT 7,S1]
000101 265300 000231*   01470   JSP   G,TS;      SEND "DATE:"  
000102 265300 000262*   01480   D2:  JSP   G,TTCR;      GET TTY CHARACTER
000103 306240 000057*   01490   CAIN  F,"/n";
000104 254000 000121*   01500   JRST  D10;
000105 306240 000015    01510   CAIN  F,15;
000106 254000 000131*   01520   JRST  D12;
000107 405240 000160    01530   ANDI  F,160;
000110 302240 000060    01540   CAIE  F,60;
000111 254000 000075*   01550   JRST  D12;
000112 712040 000005    01560   DATAI TTY,F;
000113 405240 000017    01570   ANDI  F,17;
000114 221140 000012    01580   IMULI D,D10;
000115 270140 000005    01590   ADD   D,F;
000116 301140 000144    01600   CAIL  D,D100;
000117 254000 000075*   01610   JRST  D12;
000120 254000 000102*   01620   JRST  D2;
000121 301340 000002    01630   D10: CAIL  H,2;
000122 254000 000075*   01640   JRST  D1;
000123 322140 000075*   01650   JUMPE D,D1;
000124 313147 000170*   01660   CRME  D,R1(H);
000125 254000 000075*   01670   JRST  D1;
000126 202147 000000    01680   MOVEM D,MONTH(H);
000127 402000 000003    01690   SETZM D;
000130 344340 000102*   01700   AOJA  H,D2;
000131 322140 000135*   01710   JRST  D,D14;
000132 305140 000102    01720   JUMPE D,D66;
000133 254000 000075*   01730   CAIGE D1;
000134 202140 000000    01740   JRST  D1;
000134          000000    01750   MOVEM D,YEAR;
000134          000000    01760   JRST  D1;

```

HERE WE HAVE A SLASH
BUT ONLY 2 ARE ALLOWED
AND I MUST HAVE A NUMBER > 0
CHECK RANGE
GO ASK AGAIN
SAVE MONTH OR DAY
ZERO CONVERTER

IT'S A SLASH
IT'S A CR

NON-NUMERIC -- TRY AGAIN
RECOVER CHARACTER
MASK OFF ASCII BITS

DATE COMPONENT MUST BE < 100
GET ANOTHER

CAIE F,60
JRST D12;
DATAI TTY,F;
ANDI F,17;
IMULI D,D10
ADD D,F
CAIL D,D100;
JRST D12;

JRST D2;

HERE & CR, IF NO DIGITS USE OLD VALUES
CHECK YEAR
NOT UP TO DATE

6/28/67 3431

170

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 6
 ROUTINE TO INPUT DATE AND TIME

000135 402000 000003 01780 D14: SETZM D
 000136 260740 000274* 01790 TSX LM
 000137 201040 000006 01800 MOVEI B,6
 000140 200100 000575* 01810 MOVE C, [POINT 7,S2]
 000141 265300 000231* 01820 JSP G,TS
 000142 265300 000262* 01830 GET A CHAR FROM TTY
 000143 306240 000072 01840 JSP G,TTCH;
 000144 254000 000161* 01850 CAIN F,";"
 000145 306240 000015 01860 JRST D20;
 000146 254000 000164* 01870 CAIN F,15
 000147 405240 000160 01880 JRST D22;
 000150 302240 000060 01890 ANDI F,160
 000151 254000 000135* 01900 CAIE F,60;
 000152 712040 000005 01910 JRST D14;
 000153 405240 000017 01920 DATAI TTY,F;
 000154 221140 000012 01930 ANDI F,17;
 000155 270140 000005 01940 IMULI D,4D10
 000156 301140 000074 01950 ADD D,F;
 000157 254000 000135* 01960 CAIL D,*D60;
 000160 254000 000142* 01970 JRST D14;
 000161 202140 000000 01980 TIME COMPONENT MUST BE < 60
 000162 402000 000003 02000 CONVERT TO BINARY
 000163 254000 000142* 02010 GET NEXT
 000164 202140 000000 02020 D20: MOVEM D,HR
 000165 260740 000072* 02030 SETZM D
 000166 260740 000117 02040 JRST D16
 000167 263740 000000 02050
 000168 000000 000014 02060
 000169 000000 000037 02070
 000170 000000 000014 02080 R1: DEC 12;
 000171 000000 000037 02090 DEC 31;

16:00:00 172 1967

```

021100      ; ASK ABOUT DDT AND ALLOCATE USE OF USER BLOCKS ACCORDINGLY
021200
000172 201440 000070*   MOVEI J,DATIME
000173 305440 040000*   CAIGE J,40000
000174 254000 000216*   JRST D70;
000175 260740 000274*   TSX LM
000176 201040 000012*   MOVEI B,*D10
000177 200100 000576*   MOVE C,[POINT 7,S3]
000200 265300 000231*   JSP G,TS; ASK FOR DDT
000201 265300 000262*   02200
000202 306240 000131*   02210: JSP G,TTCH;
000203 254000 000216*   02220: CAIN F,"Y"
000204 306240 000116*   02230: JRST D70; YES - DDT
000205 254000 000207*   02240: CAIN F,"N"
000206 254000 000172*   02250: JRST D60;
000207 260740 000274*   02260: D50; NO DDT
000210 201040 000006*   02270: JRST BAD CHAR - TRY AGAIN
000211 200100 000577*   02280: TSX LM
000212 265300 000231*   02290: MOVEI B,6
000213 260740 000274*   02300: MOVE C,[POINT 7,S5]
000214 201440 000067*   02310: JSP G,TS; SAY: NO DDT
000215 254000 000223*   02320: TSX LM
000216 260740 000274*   02330: MOVEI J,*DDT
000217 201040 000015*   02340: JRST D71
000220 200100 000600*   02350: TSX LM
000221 265300 000231*   02360: MOVEI B,*D13
000222 260740 000274*   02370: MOVE C,[POINT 7,S4]
000223 201100 000000*   02380: JSP G,TS; SAY: DDT AVAILABLE
000224 305440 040000*   02390: TSX LM
000225 263740 000000*   02400: D71: MCVEI C,0
000226 476002 000000*   02420: CAIGE J,40000
000227 275440 002000*   02430: DONE
000230 344100 000224*   02440: SETOM CORE(C);
000230 344100 000224*   02450: SUBI J,2000
000230 344100 000224*   02460: NOJA C,D73
000230 344100 000224*   02470:

```

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 8
ROUTINE TO INPUT DATE AND TIME

440-144-1

-162-

02490	:	WRITE AND READ CONSOLE TTY	
02500	,	SEND STRING TO CONSOLE TTY	
02510	,	B IS CHAR COUNT, C IS POINTER, F IS USED	
02520	,		
02530	,		
000231 134240 000002	TS:	ILDB F,C	
000232 712140 000005		DATAO TTY,F	
000233 712340 000010		CONSO TTY,10;	
000234 254000 000233*		JRST .-1;	WAIT FOR FLAG
000235 712200 000200		CONO TTY,200;	TURN OFF FLAG
000236 372000 000001		SOSE B;	COUNT DOWN
000237 254000 000231*		JRST TS;	MORE TO DO
000240 254006 000000		JRST (G);	RETURN
000241 422032 442564	S1:	ASCII \$DATE: \$	
000242 200000 000000		ASCII \$TIME: \$	
000243 522231 542564	02640	ASCII \$DDT?-Y/N: \$	
000244 200000 000000		ASCII \$DDT AVAILABLE\$	
000245 422112 437532	02650		
000246 545371 635100			
000247 422112 420202	02660		
000250 532031 146202			
000251 412310 500000			
000252 472364 042210	02670		
000253 520000 000000			
02680	,	OUTPUT DATE AND TIME TO CONSOLE	
000254 260740 000274*	02700	DTO: TSX LM	
000255 201040 000017	02710	MOVEI B,D15	
000256 200100 000301*	02720	MOVE C,PDAT	
000257 265300 000231*	02730	JSP G,RS	
000260 260740 000274*	02740	TSX LM	
000261 263740 000000	02750	DONE	
02760	,	READ AND ECHO ON TTY	
02770	,		
000262 712340 000040	02780	TTCH: CONSO TTY,40	
000263 254000 000262*	02790	JRST .-1;	WAIT FOR A CHAR
000264 712200 001000	02810	CONO TTY,1000;	FLAG OFF
000265 712040 000005	02820	DATAI TTY,F;	READ CHAR
000266 712140 000005	02830	DATAO TTY,F;	ECHO BACK
000267 712340 000010	02840	CONO TTY,10	
000270 254000 000267*	02850	JRST .-1;	WAIT ON OUTPUT
000271 712200 000200	02860	CONO TTY,200;	FLAG OFF
000272 405240 000177	02870	ANDI F,177;	JUST 7 BITS
000273 254006 000000	02880	JRST (G);	RETURN
02890	,	MOVE CARRIER TO LEFT MARGIN	
02900	,		
02910	,		
02920	,	MOVEI B,2	
000274 201040 000002	02930	MOVE C,[POINT 7,CRLF]	
000275 200100 000601*	02940	JSP G,TS	
000276 265300 000231*	02950	DONE	
000277 263740 000000	02960		
000300 064240 000000	02970	CRLF: OCT POINT 7,DATE;	CARRIAGE RETURN,LINE FEED AND ZERO
000301 440700 000000	02980		POINTER TO DATE

03020 ; ENTERED FOLLOWING READING JOSS FROM DRUM
03030
000302 200100 000042 RECOVER:MOVE C,42
000303 200140 000043 MOVE D,43
000304 201040 000523* MOVEI B,ACS;
000305 251040 000544* BLT B,ACS+17
000306 200040 000030 MOVE B,30
000307 202040 000526* MOVEM B,ACS+1; AND REG 1
000310 256000 000546* XCT BEG3;
000311 260740 000322* TSX CLEAR;
000312 260740 000333* TSX DISPLAY;
000313 260740 000415* TSX RESTORE;
000314 712200 000006 CONO TTY,6;
000315 700600 002402 CONO PI,2402;
000316 712300 000020 CONSZ TTY,20
000317 254000 000316* JRST .-1;
000320 260740 000165* TSX ADATE
000321 254000 000551* 03190 JRST REG5;
03200

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 10
CLEAR MACHINE STATUS

000322 201040 000007 03220 MOVEI B,7
000323 254400 000324* 03230 JRST 10,+1;
000324 367040 000323* 03240 SOJG B,-1
000325 712200 001200 03250 CONO TTY,1200;
000326 700200 111000 03260 APR,111000;
000327 722600 000000 03270 CONO MT1,0;
000330 720200 000000 03280 CONO DC,0;
000331 727200 000000 03290 CONO 270,0;
000332 263740 000000 03300 DONE
03310

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 11
OUTPUT ERROR MESSAGE AND PANEL DJMP

000333 260740 000000 033300 DISPLAY:TSX CMESS
000334 260740 000356* 03340 TSX MR0;
000335 260740 000333* 03350 TSX CMESS
03360 XMT ♦D13,M100;
AC TITLE

000336 000640 000374* X M100
000337 201240 000525* 03370 MOVEI F,ACS;
000340 260740 000335* 03380 EC1: TSX CMESS;
000341 260740 000363* 03390 TSX MR1
000342 305240 000545* 03400 CAIGE F,ACS*20
000343 254000 000340* 03410 JRST EC1
000344 260740 000340* 03420 TSX CMESS
03430 XMT ♦D50,M101;
VARIOUS à CELLS

000345 003100 000377* X M101
000346 200040 000536* 03440 MOVE B,ACS*11; REGISTER 11
000347 202040 000112 03450 MOVEM B,SA
000350 201240 000112 03460 MOVEI F,SA
000351 260740 000344* 03470 TSX CMESS
000352 260740 000363* 03480 TSX MR1
000353 260740 000351* 03490 TSX CMESS
03500 XMT 5,M102; MORE LINE FEEDS

000354 000240 000411* X M102
000355 263740 000000 03510 DONE
03520

					MAKE UP HALT NUMBER MESSAGE
000356	200240	000040	03550		
			03560	MOVE F,40;	
			03570	XMT 'D18,M102	SAVE HALT CODE
000357	001100	000411*		OPDEF X {D18B12}	
000360	200000	000005	03580		
000361	260740	000000	03590	MOVE A,F	
000362	263740	000000	03600	TSX OCTW	
			03610	DONE	
			03620		
			03630		
			03640		
000363	201300	000004	03650	MOVEI G,4	
000364	200005	000000	03660	MOVE A,(F)	
000365	260740	000371*	03670	TSX O1	
000366	350000	000005	03680	AOS F	
000367	367300	000364*	03690	SOJG G,MR1.1	
000370	263740	000000	03700	DONE	
000371	260740	000361*	03710		
			03720	O1:	
			03730	TSX OCTW	
000372	000140	000422*	X BLANKS	XMT 3,BLANKS	
000373	263740	000000	03740	DONE	
000374	406070	352632	03750		
000375	526310	152236	03760	M100: ASCII ?ACCUMULATORS?:?	
000376	512467	200000			
000377	476150	642634	03770	M101: ASCII ?OFFENDING LOCATION (11): CONTENTS: (@11): @17: @0:?	
000400	422231	643500			
000401	462370	340650			
000402	446371	620120			
000403	305425	135100			
000404	202071	747250			
000405	426352	451564			
000406	242006	130522			
000407	351010	030556			
000410	351010	030164			
000411	064241	205024	03780	M102: OCT ASCII 064241205024; CR AND 4 LFS	
000412	442031	452100	03790	?HALT NUMBER: ?	
000413	472531	541212			
000414	511644	000000	03800		

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 13
RESTORE DATE, TIME, ETC.

000415 200040 000421* 03820 RESTORE:MOVE B, RE1
000416 251040 000164* 03830 BLT B, MIN
000417 260740 000423* 03840 TSX SLM;
000420 263740 000000 03850 DONE SET LOWER MEMORY
000421 000031 000134* 03860
000422 201004 020100 03870 RE1: XWD 31, YEAR
03880 BLANKS: ASCII ? ?
03890 03900 ; SET UP LOWER MEMORY
03910
000423 200000 000602* 03920 SLM: MOVE A, [XWD LOC41, 41]
000424 251000 000137 03930 BLT A, 137
000425 263740 000000 03940 DONE
03950
03960

03980 ; 0-17 EXEC LOADER (SHADOW MEMORY)
03990 ; 20-27 PAPER TAPE LOADER - RIM
04000 ; 30 SAVE CELL FOR REG 1 DURING RECOVERY
04010 ; 31-35 SAVE CELLS FOR DATE AND TIME IN RECOVERY
04020 ; 36-37 USED BY DDT
04030 ; 40-41 DUO CELLS
04040 ; 42-57 INTERRUPT SYSTEM TRAP CELLS
04050 ; 60-77 PART OF DRUM READ ROUTINE
04060 ; 100 PSEUDO SWITCHES
04070 ; 101 DUMP ENTRY
04080 ; 102 PSEUDO CONSOLE SIGNAL CELL
04090 ; 103 ENTRY TO READ DRUM ROUTINE
04100 ; 104-137 DRUM READ AND WRITE ROUTINES
04110 ;

000426 264000 000000 04130 ; CONTENTS OF CELLS 41-137
000427 000000 000000 04140 LOC41: JSR PROP
000430 264000 000214 04150 Z
000431 264000 000000 04160 JSR DDT; 43
000432 000000 000000 04170 JSR DRMR; 44 - CHANNEL 2, THE DRUM
000433 264000 000000 04180 Z
000434 000000 000000 04190 JSR C25; 46 - CHANNEL 3, 630 INPUT
000435 264000 000000 04200 Z
000436 000000 000000 04210 JSR C26; 50 - CHANNEL 4, 630 OUTPUT
000437 254000 000430 04220 Z
000440 000000 000000 04230 JSR JRST 52 - CHANNEL 5, DISC FILE
000441 264000 000000 04240 Z
000442 000000 000000 04250 JSR CTYR; 54 - CHANNEL 6, TAPE AND CONSOLE TTY
000443 264000 000000 04260 Z
000444 000000 000000 04270 JSR APRR; 56 - CHANNEL 7, PROCESSOR
04300 Z
04310 ; DEFINITIONS FOR WHEN DRUM CODE IS MOVED TO LOW MEMORY
04320 BTW=116
04330 RJ=60
04340 RIO=72
04350 DS=73
04360 DS1=74
04370 SWITCHI=100
04380 FAKE=102
04390 RJD=103
04400 WJD=117
04410 WD=132
04420 RC=135
04430 SA=112
04440 CRD=63

04470 ; READ AND WRITE JOSS ON DRUM
 04480 04490 RJ.: CONO DP,0; 167 TO READ
 04500 JSP D,DS; READ
 04510 CONO DR,230; CHECK DRUM ERRORS
 04520 CKD.: CONSO DR,1000; TOP ERRORS
 04530 CONSZ DP,100060;
 04540 JRST 4,7770
 04550 CONSO DR,100; WAIT FOR DONE
 04560 JRST CKD
 04570 CONO DR,270; DESELECT
 04580 JRST (C)
 04590 XWD -40000,40000; I/O WORD - 16K FROM UPPER BOX
 04600 DS.: CONO DP,100; SELECT FOR WRITE
 04610 DS1.: DATAO DP,RI0;
 04620 DATAO DR,B; UNIT AND TRACK
 04630 CONO DR,260; SELECT
 04640 JRST (D)
 04650 HALTS=20000
 04660 04670 L100: Z; 100 - PSEUDO SWITCH CELL
 04680 JRST DUMP 102 - PSEUDO SIGNAL CELL
 04690 FAKE.: Z; 103 - RECOVERY ENTRANCE
 04700 RJD.: MOVEI B,0; C,RJ;
 04710 JSP C,BTW;
 04720 MOVE B,BTW; MOVE TO LOW MEMORY
 04730 BLT B,37777
 04740 MOVEI B,2000;
 04750 JSP C,RJ;
 04760 JRST RECOVER FILL UPPER BOX
 04770 SA.: BLOCK 4;
 04780 XWD 40140,140; FOUR SAVE CELLS
 04790 WJD.: MOVEI B,2000; BTW CONTROL WORD
 04800 JSP C,WD;
 04810 JSP C,RC;
 04820 MOVEI B,40000; WRITE JOSS ON DRUM
 04830 BLT B,77777;
 04840 MOVEI B,0; WRITE
 04850 JSP C,WD;
 04860 JSP C,RC;
 04870 MOVEI B,2000; READ COMPARE
 04880 JSP C,RJ;
 04890 DONE READ BACK UPPER BOX
 04900 WD.: JSP D,DS; WRITE DRUM - SELECT
 04910 CONO DR,220; WRITE DRUM
 04920 JRST CKD; GO CIICK
 04930 04940 RC.: JSP D,DS; READ COMPARE - SELECT
 04950 CONO DR,210; COMPARE DRUM AND CORE
 04960 JRST CKD
 04970 04980 ACS: BLOCK 20; TEMP STORAGE FOR ACS
 04990 05000

614-5437 181

000545 700200 210000 05020 BEGIN: CONO APR,210000;
000546 200740 000000 05030 BEG3: MOVE PP,PPW
000547 260740 000070 05040 DATETIME;
000550 712200 000206 05050 TSX CONO
000551 260740 000000 05060 TTY,206;
000552 700200 122007 05070 C20; IRWIN
000553 722200 000000 05080 SET PROCESSOR FLAGS AND CHANNEL
000554 722600 000002 05090 CONO APR,122007;
000555 700600 020000 05100 CONO MTC,0
000556 571700 777776 05110 CONO MT1,2;
000557 201040 000000 05120 CONO PI,200000;
000560 260740 000900 05130 HRREI S,-2;
000561 571700 777777 05140 MOVEI B,TYPE6
000562 260740 000000 05150 TSX SOUT
000563 254000 000000 05160 HRREI S,-1;
000564 000000 000120 TSX HMES;
000565 264000 000000 JRST SIGPR
000566 720100 000016
000567 440300 000120
000570 440300 000020
000571 440600 777700
000572 170300 000004
000573 777754 777677
000574 440700 000241
000575 440700 000243
000576 440700 000245
000577 440700 000252
000600 440700 000247
000601 440700 000300
000602 000426•000041
05170
05180
END

THERE ARE NO ERRORS

PROGRAM BREAK IS 000603

A	ACS	000000	000525*
	ADPTE	000320*	EXT
	APR	000443*	EXT
B	BEGIN	000001	
	BEG3	000546*	
	BEG5	000551*	
	BEGIN	000545*	INT
	BLANKS	000422*	
	BTW	000116	
	BTW.	000503*	
C		000002	
	C20	000551*	EXT
	C25	000433*	EXT
	C26	000435*	EXT
CAPR		000007	
CCTY		000006	
CDAT		000006	
CDC		000001	
CDRM		000002	
CH630		000030	
CHAPR		000001	
CHCTY		000002	
CHDAT		000000	
CHDC		000100	
CHDRM		000040	
CI630		000003	
CKD		000063	
CKD.		000450*	
CLEAR		000322*	
CMESS		000353*	EXT
CO630		000004	
CORE		000226*	EXT
CRLF		000300*	
CTYR		000441*	EXT
D		000003	
D1		000075*	
D10		000121*	
D12		000131*	
D14		000135*	
D16		000142*	
D2		000102*	
D20		000161*	
D22		000164*	
D50		000172*	
D51		000201*	
D60		000207*	
D70		000216*	
D71		000223*	
D73		000224*	
DATE		000301*	EXT
DATETIME		000070*	INT
DDK		000050	
DDT		000437*	EXT
DISMIS		000000*	

DISPLA	000333*	
DMT	000020	
DP	000010	
DR	000400	
DRMR	000431*	EXT
DS	000073	
DS.	000460*	
DS1	000074	
DS1.	000461*	
DTO	000234*	
DUMP	000003	INT
E	000004	
EC 1	000340*	
F	000005	
FAKE	000102	INT
FAKE.	000467*	
G	000006	
H	000007	
HALTS	020000	
HMES	000562*	EXT
HR	000161*	EXT
I	000010	
IOA	77700	
J	000011	
K	000012	
L	000013	
L100	000465*	INT
LM	000274*	
LOC4 1	000426*	
M	000014	
M100	000374*	
M101	000377*	
M102	000411*	
MIN	000416*	EXT
MONTH	000126*	EXT
MRO	000356*	
MR1	000363*	
MR 1. 1	000364*	
MT1	000224	
MT2	000230	
MTC	000220	
N	000015	
O1	000371*	EXT
OCTW	000371*	EXT
PDATE	000301*	
PP	000017	
PPW	000546*	EXT
PROP	000426*	EXT
R1	000170*	
RC	000135	
RC.	000522*	
RE1	000421*	
RECOVE	000302*	
RESTOR	000415*	
RIO	000072	

PREAMBLE AND RECOVERY 8/1/67 COPYRIGHT 1966 THE RAND CORP. PAGE 20
SYMBOL TABLE

RIO. 000457.
RJ. 000060.
RJ-. 000445.*
RJD. 000103 INT
RJD-. 000470.
S. 000016.
S1. 000241.
S2. 000243.*
S3. 000245.*
S4. 000247.
S5. 000252.*
SA. 000112.
SA-. 000477.*
SAVE. 000120.
SIGPR. 000563.* EXT
SLM. 000423.*
SOUT. 000560.* EXT
SWITCH. 000100 INT
TD05. 000022.*
TD1. 000024.*
TD2. 000030.*
TD3. 000035.*
TD4. 000037.*
TS. 000231.*
TRCH. 000262.*
TYPE6. 000557.* EXT
WD. 000132.
WD-. 000517.*
WJD. 000117 INT
WJD-. 000504.*
YEAR. 000421.* EXT

END OF ASSEMBLY

REFERENCES

1. Bryan, G. E., JOSS: User Scheduling and Resource Allocation, The RAND Corporation, RM-5216-PR, January 1967.
2. Bryan, G. E., JOSS: Accounting and Performance Measurement, The RAND Corporation, RM-5217-PR, June 1967.
3. Programmed Data Processor-6 Handbook F-65, Digital Equipment Corporation, Maynard, Mass., August 1964 (including F-65 Change Notice No. 3).
4. PDP-6 Programming Manual: MACRO-6 Assembly Language, DEC-6-0-TP-MAC-LM-F-ACT01, Digital Equipment Corporation, Maynard, Mass.

JOSS BIBLIOGRAPHYPUBLICATIONS OF CURRENT INTEREST

- Baker, C. L., JOSS: Introduction to a Helpful Assistant, The RAND Corporation, RM-5058-PR, July 1966.
- , JOSS: Console Design, The RAND Corporation, RM-5218-PR, February 1967.
- Bryan, G. E., JOSS: Accounting and Performance Measurement, The RAND Corporation, RM-5217-PR, June 1967.
- , JOSS: Introduction to the System Implementation, The RAND Corporation, P-3486, November 1966; also published by the Digital Equipment Computer Users Society, DECUS Proceedings, Fall 1966.
- , JOSS: 20,000 Hours at the Console--A Statistical Summary, The RAND Corporation, RM-5359-PR, August 1967.
- , JOSS: User Scheduling and Resource Allocation, The RAND Corporation, RM-5216-PR, January 1967.
- Bryan, G. E., and E. W. Paxson, The JOSS Notebook, The RAND Corporation, RM-5367-PR, August 1967.
- Bryan, G. E., and J. W. Smith, JOSS Language (Aperçu and Précis, Pock-et Précis, Poster Précis), The RAND Corporation, RM-5377-PR, August 1967.
- Gimble, E. P., JOSS: Problem Solving for Engineers, The RAND Corporation, RM-5322-PR, May 1967.
- Greenwald, I. D., JOSS: Arithmetic and Function Evaluation Routines, The RAND Corporation, RM-5028-PR, September 1966.
- , JOSS: Console Service Routines (The Distributor), The RAND Corporation, RM-5044-PR, September 1966.
- , JOSS: Disc File System, The RAND Corporation, RM-5257-PR, February 1967.
- Marks, S. L., and G. W. Armerding, The JOSS Primer, The RAND Corporation, RM-5220-PR, August 1967.
- Smith, J. W., JOSS: Central Processing Routines, The RAND Corporation, RM-5270-PR, August 1967.

PUBLICATIONS OF HISTORICAL INTEREST

Baker, C. L., JOSS: Scenario of a Filmed Report, The RAND Corporation, RM-4162-PR, June 1964.

"The JOSS System: Time-Sharing at RAND," Datamation, Vol. 10, No. 11, November 1964, pp. 32-36. (This article is based on RM-4162-PR above.)

04-5837 108
CAT FILE

-178-

Shaw, J. C., JOSS: A Designer's View of an Experimental On-Line Computing System, The RAND Corporation, P-2922, August 1964; also published in AFIPS Conference Proceedings (1964 FJCC), Vol. 26, Spartan Books, Inc., Baltimore, Md., 1964, pp. 455-464.

-----, JOSS: Conversations with the Johnniac Open-Shop System, The RAND Corporation, P-3146, May 1965.

-----, JOSS: Examples of the Use of an Experimental On-Line Computing Service, The RAND Corporation, P-3131, April 1965.

-----, JOSS: Experience with an Experimental Computing Service for Users at Remote Typewriter Consoles, P-3149, May 1965.

