

IBM 1620 Jr.

Diagnostic Manual

Version 0.9

CHM Computer
History
Museum



Introduction

The IBM 1620 was always shipped with a full set of diagnostic programs. These programs were typically used by the IBM Customer Engineer to check the correct operation of the machine and to diagnose / isolate / exercise failures when they occurred. They cover every aspect of the machine from console switches & lights, core memory, and instruction execution to operation of peripherals. A properly maintained IBM 1620 rarely failed, but it did happen. The diagnostics were also used when the computer was first installed to verify that it was fully operational.

While the hardware of the IBM 1620 Jr. is several orders of magnitude more reliable than the original 1960's IBM 1620, the diagnostic programs are still very useful to validate the simulation software in the main unit, the typewriter, and the card reader/punch device. Whenever changes are made to the simulators, the diagnostics should be run.

The Computer History Museum has in its collection, most of the manuals for the IBM 1620 Model 1 Diagnostics and a few paper tape images of the basic ones. Fortunately, for the diagnostics without binary images, the manuals include a simplified machine language listing of each program. One notable exception is the CU06/DT106 floating point diagnostic which is completely missing from the collection. As a replacement, a volunteer, John M. Bohn, wrote an SPS program (FP01) which thoroughly tests all of the 1620's floating point instructions. In addition, the CU02 Error Check diagnostic only tests the memory addressing register (MAR) subset of the IBM 1620's "Check Stop" conditions and depends on an internal maintenance switch, CE #9, which changed the computer's behavior. For the IBM 1620 Jr. it was replaced with a new, comprehensive error checking program (CS01).

IBM supplied all of the diagnostics on paper tape, punched cards, and the customer engineering disk pack. For the IBM 1620 Jr. all of the diagnostic programs have been "built into" the simulator for ease of use. When the IBM 1620 Jr. is powered off, pressing the RESET button loads into simulated core memory one of the diagnostic programs based on the setting of the PROGRAM SWITCHES. When the machine is turned on, pressing the START button will run the program.

Here are the programs available:

PS 1	PS 2	PS 3	PS 4	Name	Description
off	off	off	off	PowerOf2	Power of 2 Calculator Demo Program
off	off	off	on	CU00	Console Diagnostic
off	off	on	off	CU01	General Op Codes Diagnostic
off	off	on	on	CU05	Special Instructions Diagnostic
off	on	off	off	FP01	Floating Point Diagnostic
off	on	off	on	CU03	Indirect Addressing Diagnostic
off	on	on	off	CS01	Check Stops Diagnostic
off	on	on	on	DX05L	Core Storage 20K Low Diagnostic
on	off	off	off	DX05H	Core Storage 20K High Diagnostic
on	off	off	on	CU04	Additional Core Diagnostic
on	off	on	off	DX03	Typewriter Diagnostic
on	off	on	on	IO02	Card I/O Diagnostic
on	on	off	off	IO03	Card I/O Reliability Diagnostic

The following sections describe all of the diagnostic programs and how to run them.

Lights and Switches Tests

Lights and Switches Tests

Built into the IBM 1620 Jr. simulator is a way to manually test all of the lights and switches on the front panel. This is the IBM 1620 Jr.'s version of a "Lamp Test" switch found on other computers. The IBM 1620 did not have this feature, but it is a useful capability for maintenance, so it was added to the IBM 1620 Jr. Unlike the IBM 1620 diagnostic test programs, these tests do not involve executing any IBM 1620 code.

There are four different tests available. They are activated when the machine is powered off, the left-most (unlabeled) toggle switch is on, and one PROGRAM SWITCH is turned on.

PROGRAM SWITCH 1: This is the actual "Lights and Switches" test. It begins with all wired lights turned on. This tests all of the lights. As each switch is turned on or button is pressed, one or more lights are turned off. For the toggle switches [not the PROGRAM SWITCHes], the light(s) above it are turned off when the switch is on. The left-most toggle switches and the PROGRAM SWITCHes are checked by selecting the various tests. The button / light correspondence is:

RESET: POWER ON, POWER READY, THERMAL lights

DISPLAY MAR: Pressing this button tests the large MEMORY ADDRESS REGISTER DISPLAY SELECTOR rotary switch. The HUNDREDS row of the MEMORY ADDRESS REGISTER displays a binary value representing the position of the knob as it is rotated, where: OR-1 = 0, OR-2 = 1, OR-3 = 2, ..., IR-2 = 11

SAVE: SAVE light

INSERT: INSERT light

RELEASE: PUNCH NO FEED, READER NO FEED lights

START: AUTOMATIC light

STOP / SIE: MANUAL light

INSTANT STOP / SCE: CHECK STOP light

PROGRAM SWITCH 2: This test displays the full LED brightness range. Each column of lights in the upper front panel displays a different brightness level from off in the left-most column to fully on in the right-most column. The columns in-between are 36 of the 254 intermediate brightness levels available.

PROGRAM SWITCH 3: This "snake" demo moves multiple lights across the upper and lower front panels. The two lights in the upper panels move randomly. The two lights in the lower panels move back and forth across the available wired lights. The speed of animation is controlled by the MEMORY ADDRESS REGISTER DISPLAY SELECTOR rotary switch, where PR-3 is the fastest and OR-4 is the slowest.

PROGRAM SWITCH 4: This one is just for fun. It lights "IBM 1620" in the upper front panels.

One thing to note about the IBM 1620 Jr.'s front panel – not all of the lights are wired and therefore they cannot be turned on. This is due to a combination of design decisions made when Jr. was built. None of the inactive lights were used on a real IBM 1620 Model 1 Level F machine, so it does not affect correct operation of the IBM 1620 Jr. However, the unlit lights are noticeable in the “lights and switches” tests. The inactive lights are:

INSTRUCTION AND EXECUTE CYCLE: E-24, E-25

CONTROL GATES: SET 00080, VRC GATE, END COMPR

INPUT-OUTPUT: *row 1 light 3, row 3 light 1, row 5 light 1*

DIGIT REGISTER: *row 1 light 2, row 2 light 2*

Lower Left Panel: *row 1 light 1, row 1 light 5, row 2 light 1, row 3 light 1*, MBR-E CHK, MBR-O CHK

Status Lights: *light 8, light 11, light 13*

CU00 – Console Diagnostic


```
=====  
//  
// CU00 - Console Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: not used  
// PS2: not used  
// PS3: not used  
// PS4: not used  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - STOP  
//  
// Start addresses:  
//  
// 00000 - Full test  
//  
// Directions:  
//  
// 1. Load CC diagnostic  
// 2. Press RESET  
// 3. Press START  
// 4. After printing the diagnostic program, the machine will halt with the  
//    MANUAL light on  
// 5. Verify the register contents:  
//    OPERATION REGISTER - 48  
//    MULTIPLIER - 0  
//    SENSE AND BRANCH - 0  
//    MEMORY BUFFER REGISTER - 22  
//    MEMORY DATA REGISTER - 2  
//    DIGIT REGISTER - 22  
//    MEMORY ADDRESS REGISTER - 11111  
// 6. Press SAVE  
// 7. Verify the MEMORY ADDRESS REGISTER is 11112  
// 8. Turn the MEMORY ADDRESS REGISER DISPLAY SELECTOR, press DISPLAY MAR, and  
//    verity the MAR register contents:  
//    IR-1 - 11112  
//    IR-2 - 00000  
//    OR-1 - 02222  
//    OR-2 - 04444  
//    OR-3 - 04444  
//    PR-1 - 11112  
//    PR-2 - 00000  
//    PR-3 - 00000  
// 9. Press INSERT  
// 10. Type 4911112  
// 11. Press RELEASE-START  
// 12. Verify AUTOMATIC light is on and the MANUAL light is off  
// 13. Type ABCDEFGHIJKLMNOPQRSTUVWXYZ <record mark>  
// 14. Press RELEASE  
// 15. Press START  
// 16. After printing the alphabet, the machine will wait for alphameric input  
// 17. Verify the AUTOMATIC light is on and the SAVE light is off  
// 18. Verify the MEMORY ADDRESS REGISTER is 11123  
// 19. Press RELEASE
```

```
// 20. Verify the AUTOMATIC light is off and the MANUAL light is on
```

```
//
```

```
//=====
```

Sample Output - CU00

38ØØØØØØ1ØØ3111ØØØØ364911ØØØØØ48Ø444Ø222237Ø5555ØØ1ØØ39Ø5555ØØ1ØØ42ØØØ84Ø
ØØØØ
4911112§
ABCDEFGHIJKLMNOPQRSTUVWXYZ#
ABCDEFGHIJKLMNOPQRSTUVWXYZ

NO. 2128300
SHEET 0
OF 3
0

DIAGNOSTIC TEST

TITLE 1620 DATA PROCESSING SYSTEM INDEX AND CONSOLE CHECK-OUT
MACH. TYPE 1620 BY J. H. M. APPR. G. I. A. DATE 4-11-62

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404530	8-15-60	1, 2
404568	12-15-60	1, 2, 3
404618	5-15-61	1
404675	4-11-62	1, 2, 3
404890-G	9-17-63	1, 2, 3, 4
404980	5-7-64	1, 2, 3
412514	6-28-64	1
412553	21-SEP-65	1

E/C NO.	404980	412514	412553				
DATE	5-7-64	6-28-64	21-SEP-65				

1620 DATA PROCESSING SYSTEM DIAGNOSTICS

INDEX

Name	Description	P/N Manual Page	P/N Paper Tape	P/N Card Deck
CU01	*General Op Codes	1620	2128301	2128302
CU02	*Error Check Test	1620	2128303	2125687
CU03	Indirect Addressing Feature	1620	2125574	2125688
CU04	Additional Core Feature	1623	2125704	2125689
CU05	Special Instructions Feature	1620	2125637	2125639
CU06	Floating Point Reliability Test	1620	2153435	2153436
DX01	*Header Test	1620	2172330	2172332
DX02	*CPU Test	1620	2172333	2172335
DX03	*Typewriter Test	1620	2172336	2172338
DX04	Special Instructions Test	1620	2172339	2172341
DX05L	*Core Storage Test L-20K	1620	2172342	2172344
DX05H	*Core Storage Test H-20K	1620	2172334	2172345
DX06	Indirect Addressing Test	1620	2172346	2172348
DX07L	Core Storage Test L 40/60K	1623	2172349	2172350
DX07H	Core Storage Test H 40/60K	1623	2172360	2172352
DT106	Floating Point Feature	1620	2158959	2158990
I002	Card I/O Feature	1622	2125684	-
I003	Card I/O Reliability Test	1622	2125682	-
DT20	Write Addresses	1311-3	2161816	2161818
DT21	Write Test Data	1311-3	2161819	2161821
DT22	Fault Isolation	1311-3	2161822	2161824
DT43	Printer Test	1443	2172153	2172154
DT44	Plotter Test	1626	2161849	-
Dipal	Dipal Monitor	1620	2172363	2172365
DT09	*Meter Verification Test	1620	2172353	2172354

* SUPPLIED WITH BASIC SYSTEMS. OTHER DT'S SUPPLIED ONLY TO SYSTEMS INCORPORATING OPTIONAL FEATURE.

1620 CONSOLE CHECK OUT

Loading a routine requires proper console and computer unit operation. A quick check of the console operations prior to running the diagnostic tests can be made and is desirable. A suggested procedure for the check out of the console will be given later. The proper execution of these instructions will indicate that valid information can be stored into memory and then read out, that portions of MARS are functioning, and that the console is operating correctly.

To load information into the memory from an input device, it is necessary to execute a read operation (36 or 37). For 1621 input, this operation must be written into memory from the typewriter keyboard. For 1622 I/O, this operation is accomplished by pressing the 1622 load key. The input device must then be placed in a ready condition and the read operation executed. Thus the necessity to check the console.

The console operations to be checked are: Reset, Insert, Release, Read Numeric from Typewriter, Read Alpha from Typewriter, Save, Write Numeric with Typewriter, Write Alpha with Typewriter, Display MARS.

The console check out will consist of keying in a set of instructions, one of which will cause the keyed in information to be typed out. (This will be a check of the read and write numeric information.) A branch operation will then be executed. The instruction branched to will be a halt command. Depression of the Save key will then cause the contents of IR-1 to be stored in PR-1 without replacing the information in IR-1. (IR-1 is now blank.)

The operation of Display MAR can now be checked by setting the indicator switch to the desired register and depressing the Display MAR switch. To continue, insert the instruction branch to 11112. Release and Start. The machine will stop, calling for a Read Alpha from the keyboard. Key in the alphabet, followed by a record mark. Then Release and Start. The alphabet will then be typed out, the branch back instruction executed and the machine will halt with 11123 in MAR.

SUGGESTED CONSOLE CHECK PROCEDURE

1. Reset (Do not reset again during this check procedure)

2. Insert

3. Key in

38 00000 00100
31 11100 00036
49 11100 00000
48 04444 02222
37 05555 00100
39 05555 00100
42 00084 00000 ≠ (MAR should read 00084)

4. Release and Carriage return

1620 Console Check Out

5. Start. The data keyed in should be typed out. (All but record mark). The instructions TR, B, H, WA, BB should be transferred to memory positions 11100 and up. The program should HALT, and the MANUAL light turned on. The following readings should be noted: MAR-11111, MBR-22, MDR-2, DR-22, OP-48, and Sense and Branch CC on machines prior to E suffix; MAR-11111, MBR-22, MDR-2, OP-48, and Digit and Branch-22 on E suffix and later machines.
6. SAVE. MAR should read 11112.
7. Check display MARS (CE SW 9 Must be on, see note)
IR-1 blank
OR-1 C2222
OR-2 C4444
OR-3 C4444
PR-1 11112
8. Insert
9. Key In 49 11112 00000
10. Release
11. Start. The machine will stop; call for a Read Alpha from the keyboard.
12. Key in the alphabet followed by a Record Mark.
13. Release
14. Start. The alphabet will be typed out, the program will Branch Back to 11112, and stop with a call for Read Alpha from the keyboard. The save light should have been turned off. MAR will read 11123.

The machine having performed these operations properly, the check-out of the 1620 can be continued with CU01 and CU02 with a degree of confidence that data can be written into and read out of memory.

Note: 1. With CE SW 9 on, the CE remote Start key must be used.
2. On A suffix machines there is no CE SW 9; set MAR Check SW to "PROGRAM".

CU01– General Op Codes Diagnostic


```
//=====
//  
// CU01 - General Op Codes Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Bypass error type out  
// OFF - Type out routine number on error  
// PS2: ON - Loop in routine  
// OFF - Continue to next routine  
// PS3: ON - Stop on error  
// OFF - Do not stop on error, continue  
// PS4: ON - Repeat test CU01  
// OFF - Run test CU01 once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - PROGRAM  
//  
// Start addresses:  
//  
// 00828 - Full test w/o automatic divide  
// 14004 - Full test w/ automatic divide  
//  
// Directions:  
//  
// 1. Load CU01 diagnostic  
// 2. A tab stop is automatically set at column 10  
// 3. Press START  
// 4. Load blank cards in card punch  
// 5. Press START  
// 6. After 1620 stops, move card deck from punch to reader  
// 7. Press START  
//  
//=====
```


Sample Output - CU01

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR CUØ1 THEN START

START ROUTINES. ETOS FOLLOW.

12345 6789Ø
12345 6789Ø 12345
12345 6789Ø

NUM INFO ABOVE OFFSET TO RIGHT TWO SPACES BETWEEN 5 AND 6 THREE LINES OF DATA

19976Ø123456789#Ø1Ø199989

TEST ROUTINES COMPLETED. IF SW1 OFF AND NO ROUTINE NOS TYPED OUT, MACHINE PERFOR
MED TESTS PROPERLY.

19976Ø123456789Ø1Ø199989
.Ø+\$Ø-/, Ø=@ ABCDEFGHIJKLMNOPQRSTUVWXYZØ123456789
.Ø+\$Ø-/, Ø=@ ABCDEFGHIJKLMNOPQRSTUVWXYZØ123456789
.Ø+\$Ø-/, Ø=@ ABCDEFGHIJKLMNOPQRSTUVWXYZØ123456789

NO. 2128301
SHEET 0
OF 64

DIAGNOSTIC TEST

TITLE 1620 (BASIC MACHINE AND AUTOMATIC DIVISION) DIAGNOSTIC TEST - CU01

MACH. TYPE 1620 BY J. H. M. APPR. G. I. A. DATE 4-11-60

49 14004

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404530	8-15-60	1-64
404568	12-15-60	4, 12, 14, 25, 30, 60, 62, 64
404618	5-15-61	1, 1A, 6A, 12, 13, 36, 38, 55, 55A, 56, 63, 63A, 64
404675	4-11-62	1, 1A, 2, 3, 4, 6A, 11, 11A, 12, 15, 17, 24, 26, 27, 28, 29, 30, 31, 32, 33, 41, 42, 43, 46, 55, 55A, 56, 62, 62A, 63, 63A, 64

E/C NO.	404530	404568	404618	404675			
DATE	8-15-60	12-15-60	5-15-61	4-11-62			

1620 DIAGNOSTICS

Test CU01

A. SCOPE:

This test is essentially a fault detection test designed to check for proper functioning of all standard operation codes, the optional feature DIVIDE operation codes, and the circuitry associated with these codes. Failure of an operation to function properly will cause the associated routine number to be typed out (provided Switch #1 is OFF). However, since the operation code and E time triggers are known for each routine, a failure of one routine will indicate the circuits or components that may be the source of the trouble. Failure of several routines may establish a pattern that will further isolate the failing component(s).

B. SET UP:

Seven switches must be set for the desired operation. These are the three check switches, Data Check Switch, Overflow Check Switch, I/O Check Switch. If set in the PROGRAM position, a check error will not cause a machine stop; only the light will be turned on and the indicator can be interrogated and turned off by the running program. If set in the STOP position, the program will halt at the end of the memory cycle which causes the indicator to turn on. The suggested settings for these switches when running CU01 are Data Check - Stop; I/O - Stop; Overflow - Program. On suffix A machines, there is no I/O Check Switch; there is a MAR Check Switch which should be set to stop.

The four console Sense switches have the following functions in this test and should be set as desired: (SUGGESTED SETTING, ALL SWITCHES OFF)

SWITCH #1	ON	-	Bypass error type out
	OFF	-	Type out routine number on error
SWITCH #2	ON	-	Loop in routine
	OFF	-	Continue to next routine
SWITCH #3	ON	-	Stop on error
	OFF	-	Do not stop on error, continue
SWITCH #4	ON	-	Repeat Test CU01

NORMAL LOAD FROM TAPE READER:

To run the entire test, the paper tape containing the memory load for CU01 must be loaded in the paper tape reader and the reader put in a REEL mode and a READY condition. Also, put the tape punch in the READY condition. The following instruction must be manually inserted in memory locations 00000-00018: 3600024003004900828 Then RELEASE and START.

NORMAL LOAD FROM CARD I/O

Place card deck for CU01 in read hopper. Load and run in blank cards in punch hopper. Reset 1620. Depress Load key; which will cause the core storage to be loaded for CU01.

Routines with instructions addressing the paper tape reader and paper tape punch are the only routines that are different for card I/O. These differences are minor; i. e., a change to select card reader or card punch instead of paper tape reader or paper tape punch, and a change of P addresses to account for the 80 character positions of the card.

The first eight cards contain loading instructions and the math tables. The first two cards contain 8 instructions. The second through sixth instructions load the math tables at 60 characters per card. The seventh instruction loads core storage positions 00000 to 00060 with the loading instructions and the eighth instruction branches to 00000.

The loading instructions are:

```
11 00030 00060  
14 00030 19944  
36 00384 00500  
47 00000 01200  
49 00828 00000
```

The compare instruction in the above routine detects when core storage has been loaded for CU01. An equal comparison indicates that the core storage load is complete, and the program then branches to the routine that types out the setting of the sense switches.

The interlock circuits of the card reader are such that when the read hopper is empty the machine will stop on the next command for a card read. Two cards remain in the read feed. To transport these cards past the read brushes and transmit the data to the 1620 core storage, the 1622 start key must be depressed or two blank cards placed behind the deck when the deck is placed in the hopper. One of these two methods MUST be used to complete the core storage load for CU01 and to commence with the execution of the routines.

CU01

The first program executed by CU01 is a check of the console sense switches. The setting of these switches are typed out along with the instruction to set these as desired; then press the START key; HOWEVER, to check DIVIDE if installed, the following operations must be performed BEFORE depressing START:

1. INSERT
2. KEY IN 4914004
3. RELEASE
4. START

These operations will cause the instruction, branch to first division routine, to be written in memory positions 13992 - 13998, and then branch to 00552.

Division is an optional feature; although the routines to check out division are included in the program. If division is not installed, the load dividend and the divide OP codes are not valid and would cause the machine to "hang-up" when entered in the OP register and an execution attempted. Thus, a branch operation to skip the division routines is included in the program. This instruction must be altered to check division.

ENTER SINGLE ROUTINE FROM KEYBOARD

A single routine can be entered from the keyboard in the following manner:

1. Manually insert the instructions 36xxxxx0010049yyyyy from the keyboard. (xxxxx is the first memory location of the routine. In most cases it is the first position of the constant or working area. yyyy is the first instruction of the subroutine.)
2. Then release and start. The machine will "hang-up" waiting for information from the keyboard.
3. Key in the constants and instructions of the routine. Then release and start.
4. With Switch #2 on, the machine will loop in this routine.
NOTE: If the instruction involves arithmetics, the arithmetic tables must be loaded in memory locations 100-399.

1620 DiagnosticsCU01PRODUCE NEW PAPER TAPE:

To regenerate or produce another tape for input, read in the MASTER tape. When the machine halts after typing out the status of the program sense switches, Reset, Insert, key in the instructions 35000240020048, Release, Start.

C. TEST METHOD:

This test is made up of a number of sub-routines. Each sub-routine checks an operation code for specific condition and can be run as an individual test. Each sub-routine has associated with it constants and a working area, the test routine, and an error routine; and these take a block of memory. No other routine will use this block of memory. The only exception is arithmetics, where the add and multiply tables and the product area are involved.

The test was designed to first check out the more simple decision elements to determine their proper operation. As a decision element was proved to be working correctly, it was used to check the next more complicated routine.

The test starts with Checking out Branch No Record Mark, then proceeds to checkout Branch No Flag, Branch on Digit, Branch Indicate, Branch No Indicate, Transmit Digit, Transmit Digit (Immediate), Transmit Field, Transmit Field (Immediate), Transmit Record, Branch and Transmit, Branch Back, Branch and Transmit (Immediate), Set Flag, Clear Flag, Add, Add (Immediate), Subtract, Subtract (Immediate), Compare, Compare (Immediate), Multiply, Multiply (Immediate), Control, Write Numerically, Write Alphanumerically, Dump Numerically. Load Dividend, Load Dividend (Immediate), Divide, Divide (Immediate) are checked just prior to the control check if the instructions are followed.

Routines in CU01, other than those that check Input-Output operations, are performed 1000 times. This is done to give a good exercise to the logic and to have the program run for an interval of time that can be visually noted. The 1000 repeat takes in the order of 150 seconds (without division).

Upon completion of the 1000th loop, the program enters the routines (077-079) for checking the Input-Output functions.

Routine 077 checks carriage return, tab, space, write numeric, and write alphanumeric operations.

Routine 078 checks dump numeric, and routine 079 checks write alpha on cards or paper tape. After these routines are completed, the machine enters the "completed test" routine. The machine will halt if Switch #4 is OFF.

1620 Diagnostics
CU01

The typeout of routines 077 - 079 should appear as follows:

12345 67890
 12345 67890 12345
 12345 67890

NUM INFO ABOVE OFFSET TO RIGHT TWO SPACES BETWEEN
 5 AND 6 THREE LINES OF DATA.

199760123456789~~12~~199989

The characters that appear on the left margin and the length of the lines will depend upon the setup of the typewriter. The first tab stop should be at least ten characters from the left-hand margin.

To check the paper tape output, load paper tape in reader after system has come to a HALT after performing CU01. Select the STRIP mode. START. The output tape will then be read into memory and typed out. The dump numeric information should be identical to that which was dumped to the typewriter except that the record work is omitted. Three identical groups of write alpha data will be typed out.

199760123456789~~12~~199989
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

Routine 097 checks for proper operation of the arithmetic indicators and BI and BNI on these indicators.

Typical error typeouts are H followed by the routine number. If the first digit following the H is a 5, this indicates that it is a sub-routine associated with the routine designated by the last two digits; i. e., (H529 or H016). H529 is a subroutine of routine 29.

The complete normal typeout information will be as follows: (Note: The numbers after "THEN START" are present only if DIVIDE is installed and checked.)

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR CU01.
 THEN START. 4914004

START ROUTINES. ETOS FOLLOW.

12345 67890
 12345 67890 12345
 12345 67890

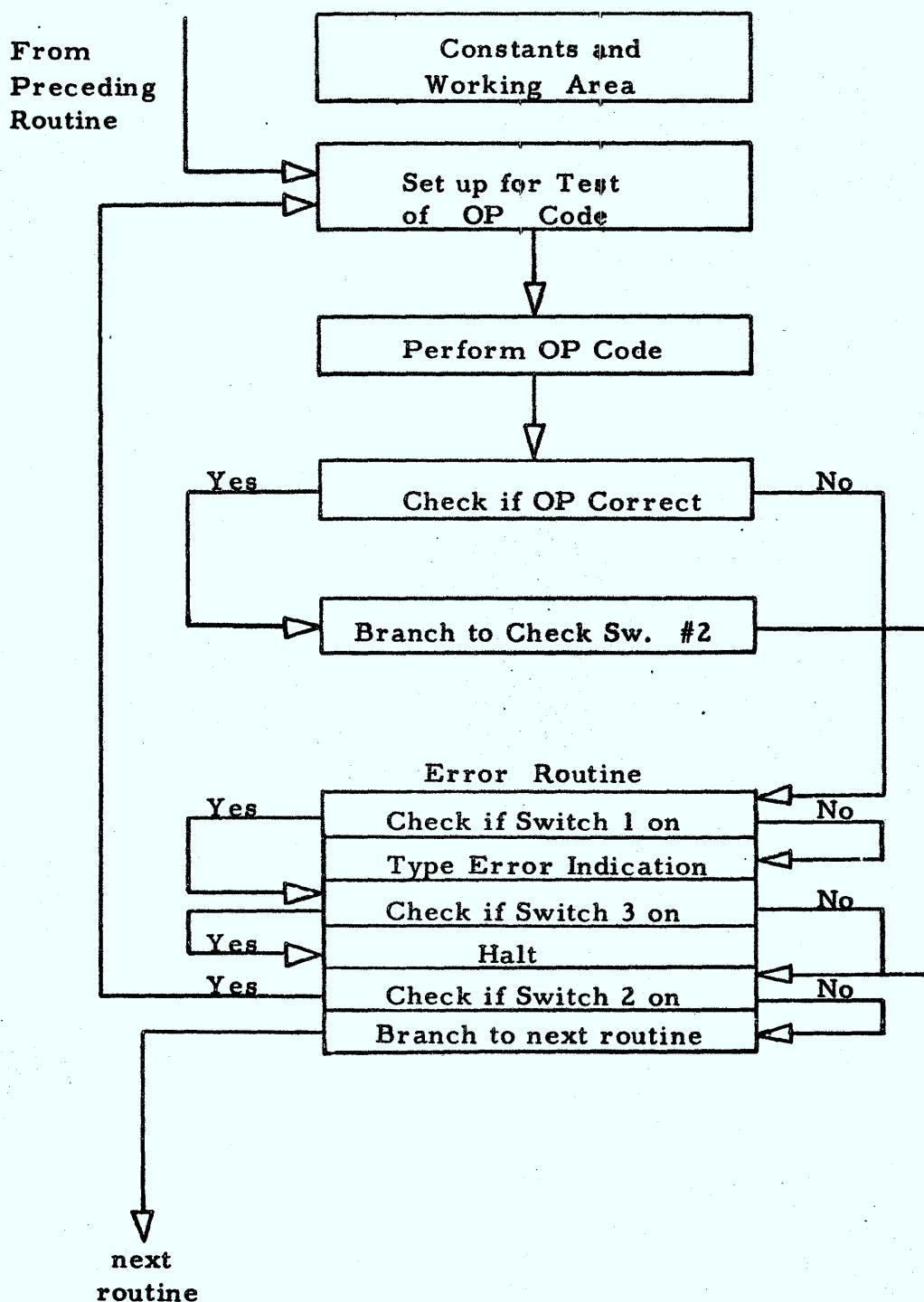
NUM INFO ABOVE OFFSET TO RIGHT TWO SPACES BETWEEN 5 AND
 6 THREE LINES OF DATA

199760123456789~~12~~199989
 TEST ROUTINES COMPLETED. IF SW1 OFF AND NO ROUTINE NOS
 TYPED OUT, MACHINE PERFORMED TESTS PROPERLY.

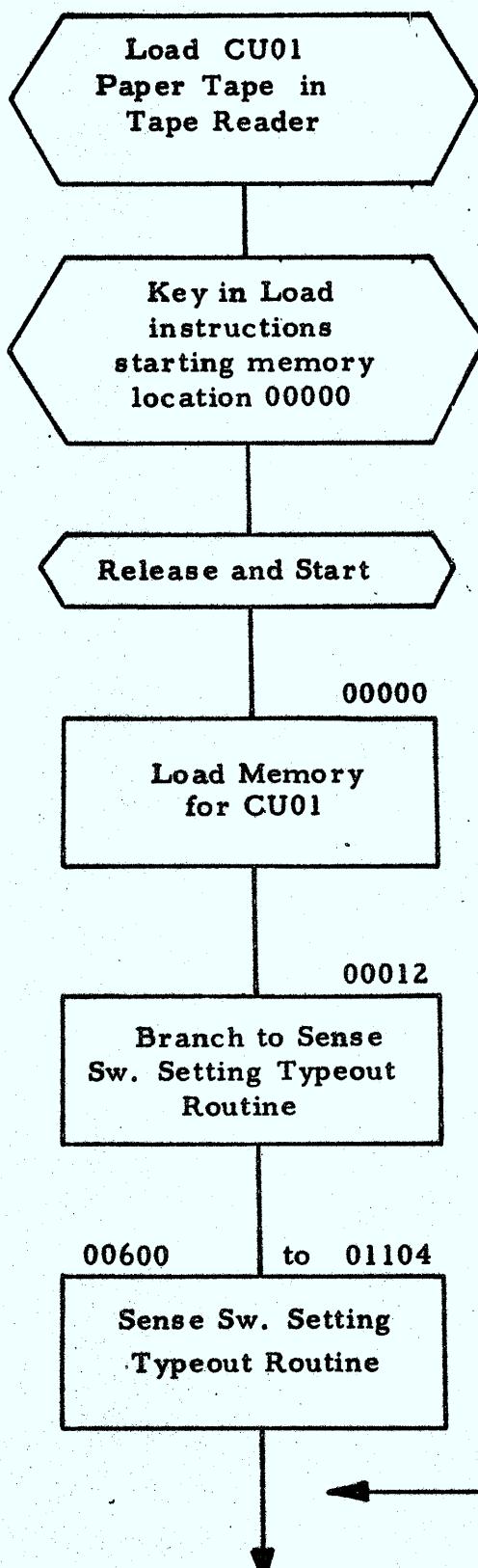
19976012345678912199989
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
 .)+\$*-/, (=@ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

CU01

Typical Flow Chart of a Test Routine



CU01 FLOW CHART



Load
Instructions are
360002400300
4900828

This instruction will load
all of memory starting at
00024. A # will be loaded
in 00000.

E Cycle Trigs

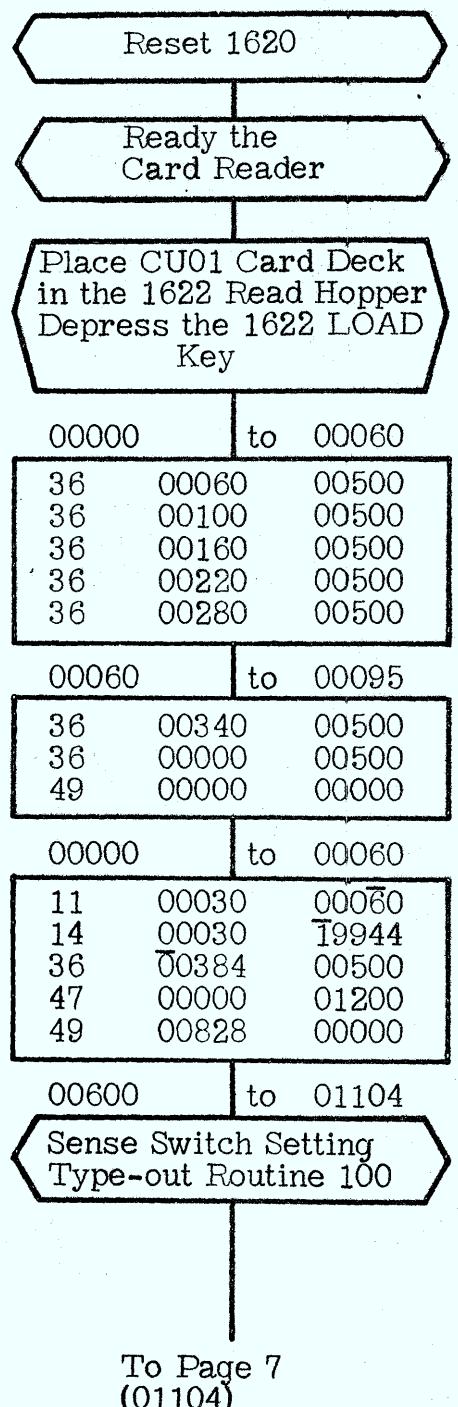
18

19

This routine will indicate if BI and/or
BNI codes are working. Machine will
HALT with 01091 in MAR. START
must be depressed to resume.

If DIVIDE installed, modify
BRANCH instruction at 13992
to check. DIVIDE. Key in
3613992001004900552. Release
and start. Key in 4914052,
Release and Start.

CU01 FLOW CHART
FOR 1622 I/O



First and Second Load Cards load the math tables and the Program Load Card (Card #8). (Cards #3 through #7 contain the math tables.)

Card #2

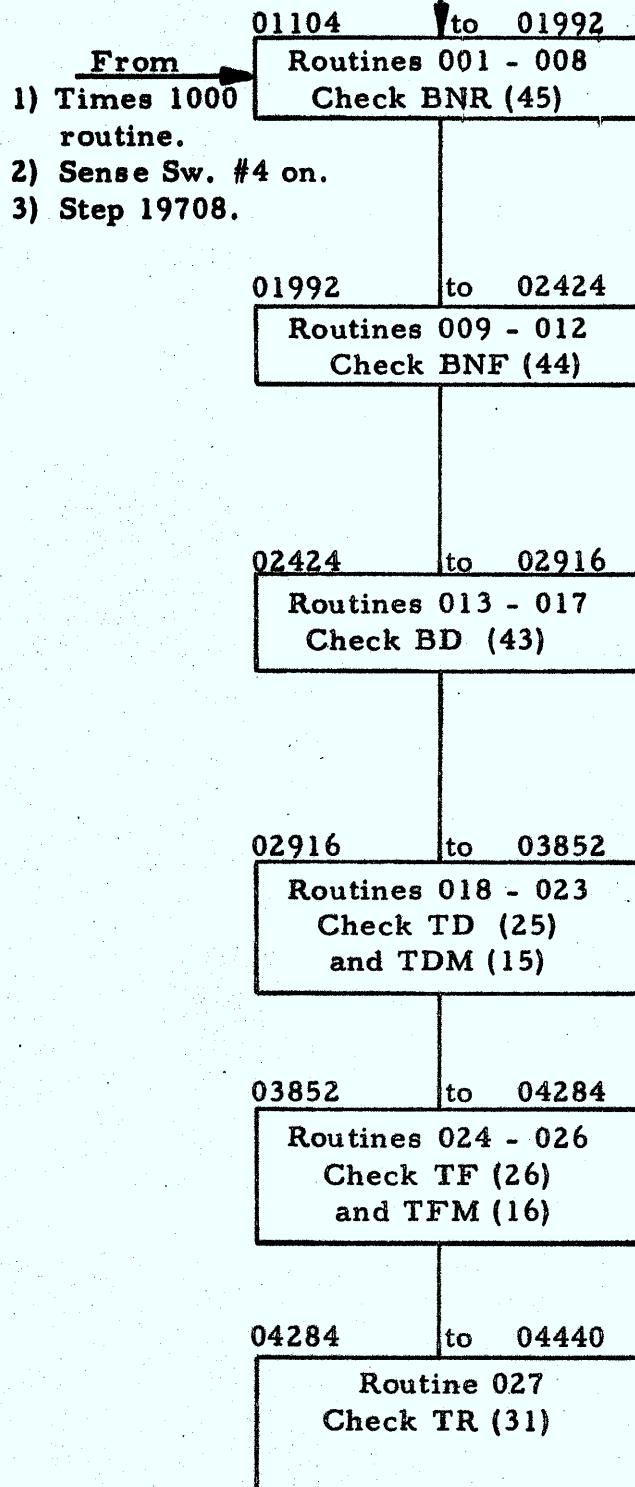
Load Card #8 contains instructions for loading the CU01 routines.

Card #8

This routine will indicate whether BI and/or BNI are working. The 1620 will Halt with 01091 in MAR. To check DIVIDE, if installed, see Page 2. If DIVIDE is not installed, press Start.

To Page 7
(01104)

From Page 6
(01091)



E Cycle Trigs. Aux. Trigs.

28 Check for RM RM
29
18 Branch
19

E Cycle Trigs. Aux Trigs.

28 Check for FLAG FM #1
29
18 Branch
19

E Cycle Trigs. Aux. Trigs.

28 Check for digit Digit
29
18 Branch
19

E Cycle Trigs. Aux. Trigs.

26 Read Mem. None
27 Write Mem.

E Cycle Trigs. Aux. Trigs.

26 Read Mem. First Cycle
27 Write Mem. Decr. FM#

E Cycle Trigs. Aux. Trigs.

26 Read Mem. Incr.
27 Write Mem. RM

To Page 8
(04440)

PN 2128301
EC 404530

From Page 7

(04284)

04440 to 05004

Routines 028 and 029
528 and 529
Check BT (27)
BTM (17) & BB (42)

E Cycle Trigs.

BT & BTM
15 Set up IR-2
16 Set up IR-1
26 } Transmit
27 } Field

Aux. Trigs.

BT & BTM
Decr.
First Cycle
FM #1

BB
20
19

BB
Save control
status

05004 to 05604

Routines 030 - 033
Check Set Flag (32)

E Cycle Trigs.

28
29

Aux. Trigs.

None

05604 to 06228

Routines 034 - 037
Check Clear Flag (33)

E Cycle Trigs.

28
29

Aux. Trigs.

None

06228 to 07248

Routines 038 - 042
Check H/P, E/Z
Trigs.; ADD (21)

E Cycle Trigs.

11
12
13
14
23

Aux. Trigs.

H/P, Carry In
E/Z, Carry Out
T/C, Recomp.,
#22; Incr./Decr
First Cycle
FM #1
FM #2

07248 to 09024

Routines 043 - 050
Check H/P, E/Z
Trigs.; Subt. (22)

E Cycle Trigs.

11
12
13
14
23

Aux. Trigs.

H/P
E/Z
T/C
Incr./Decr.
First Cycle
FM #1
FM #2
Carry Out
Carry In
Recomp., #22

To Page 9
(09024)

PN 2128301
EC 404530

**From Page 8
(09024)**

09024 to 09924

Routines 051 - 055
Check for Correct
Memory Look Up on
Add (21); Sub. (021)

E Cycle Trigs.

11
12
13
14
23

Aux. Trigs.

H/P
E/Z
T/C
FM #1
FM #2
Carry Out
Carry In
Incr. / Decr.
First Cycle
Recomp., #22

09924 to 10596

Routines 056 - 059
Check Off Trig. On
Add (21), Sub. (22),
Add (11), SM (12)

E Cycle Trigs.

11
12
13
14
23

Aux. Trigs.

H/P #22
E/Z O'Flow
T/C
FM #1
Carry Out
Carry In
Incr. / Decr.
Recomp.

10596 to 11064

Routines 060 - 062
Check Comp. (24)
For H/P

E Cycle Trigs.

Depends upon length
and sign of fields.
If all trigs. used:
11
12
13
14
21

Aux. Trigs.

T/C, H/P, E/Z,
Incr. / Decr.
First Cycle
FM #1
FM #2
Carry Out
Carry In

11604 to 11544

Routines 063, 064
Check Comp. (24)
For E/Z

E Cycle Trigs.

Depends upon length
and sign of fields.
If all trigs. used,
they are:
11
12
13
14
21

Aux. Trigs.

T/C, H/P, E/Z,
First Cycle,
FM #1
FM #2
Carry Out
Carry In
Incr. / Decr.

From Page 9

(11544)

11544 to 11712

Routines 065, 066
Check Comp. (24)
For Not H/P,
Not E/Z

E Cycle Trigs.Depends upon length
and sign of fields.

If all used:

11
12
13
14
21Aux. Trigs.

T/C, H/P, E/Z,
First Cycle
FM #1
FM #2
Incr./Decr.
Carry Out
Carry In

11712 to 11868

Routine 067
Check Comp. Immed.
(14) for E/Z

E Cycle Trigs.11
12
13
14Aux. Trigs.

T/C, H/P, E/Z
First Cycle
FM #1
FM #2
Incr./Decr.
Carry Out
Carry In

11868 to 12228

Routines 068 - 069
Check Add (21) and
Subt. (22)-Comp. Ans

E Cycle Trigs.11
12
13
14
21Aux. Trigs.

E/Z, H/P, T/C
First Cycle
FM #1
FM #2
Incr./Decr.
Carry Out
Carry In

12228 to 13272

Routines 070 - 076
Check Multi (23)
and Multi, Immed(13)
Comp. Ans.

E Cycle Trigs.32
33
34
35
36
37
38
39
40
41
19Aux. Trigs.

E/Z, H/P
T/C
First Cycle
Carry Out
Carry In
FM #1
FM #2
Incr./Decr.
Cycle Control
00080

From Page 10
(13272)

13992 to 14028

Bypass Divide
Programs Unless
this instruction
changed to
(49 14052)

Bypass Divide

If Divide installed and branch to
14052 has been keyed into 13992,
Program will branch to 14052.

14052 to 14620

Routines 080 - 082
Check Load
Dividend(28) and Load
Dividend Immed.(18)

E Cycle Trigs.

32
26
27
28
29

Aux. Trigs.

H/P
Incr. / Decr.
00080
First Cycle
Cycle Control
FM #1
FM #2
Dividend and
Remainder Sign

14620 to 15568

Routines 083 - 087
Check Divide (29)
Divide Immed. (19)

E Cycle Trigs.

32 13
26 14
27 21
28 42
29 43
11 44
12 45

Aux. Trigs.

Incr. / Decr.
00080
First Cycle
H/P; Cycle Con.
FM #1; FM #2
Dividend and Re-
mainder Sign,
First Divide
Cycle; T/C
Divide add
Carry Out
Carry In
Last Div. Cycle

15568 to 15844

Routines 088 and 089
Check Divide
By Zero Indication
and O/F Indication

E Cycle Trigs.

32 13
26 14
27 21
28 42
29 43
11 44
12 45

Aux Trigs.

Incr. / Decr.
00080
First Cycle
H/P; Cycle Con.
FM #1; FM #2
Dividend & Re-
mainder sign;
First Divide
Cycle; T/C
Divide Add
Carry Out
Carry In
Last Div. Cycle

To Page 11A
(18808)

PN 2128301

EC 404675

From Page 11
(15844 or 14028)

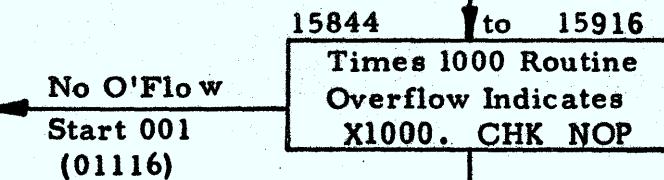
18808

19312

Routine 097
Check BI (46) and BNI (47)
on H/P, E/Z, H/P or E/Z,
and O/F Indicators

To Page 12
(15844)

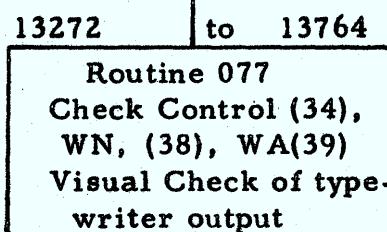
From Page 11A
(19312)

E Cycle Trigs.

11	18
12	19
13	
14	
21	

Aux. Trigs.

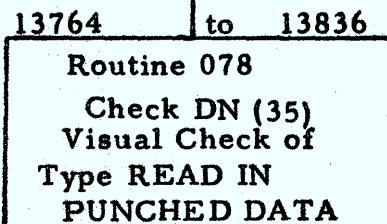
Branch Test
T/C, H/P, E/Z
First Cycle
FM #1; FM #2
Incr./Decr.
Carry Out
Carry In

E Cycle Trigs.

30
31

Aux. Trigs.

I/O Hold
RSP GATE
R/W Call
Discon. Gate
I/O Exit
I/O Sync.

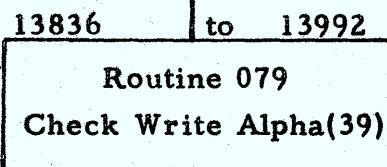
E Cycle Trigs.

30
31

Aux. Trigs.

I/O Hold
RSP Gate
R/W Call
Discon. Gate
I/O Exit
I/O Sync.
19999 Stop

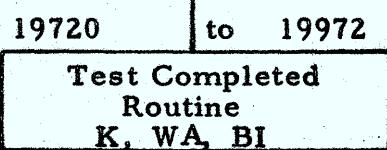
Data for DN Test
19976 - 19999

E Cycle Trigs.

30
31

Aux. Trigs.

I/O Hold
RSP Gate
R/W Call
Discon. Gate
I/O Exit
I/O Sync.

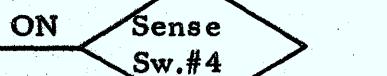
E Cycle Trigs.

30
31

Aux. Trigs.

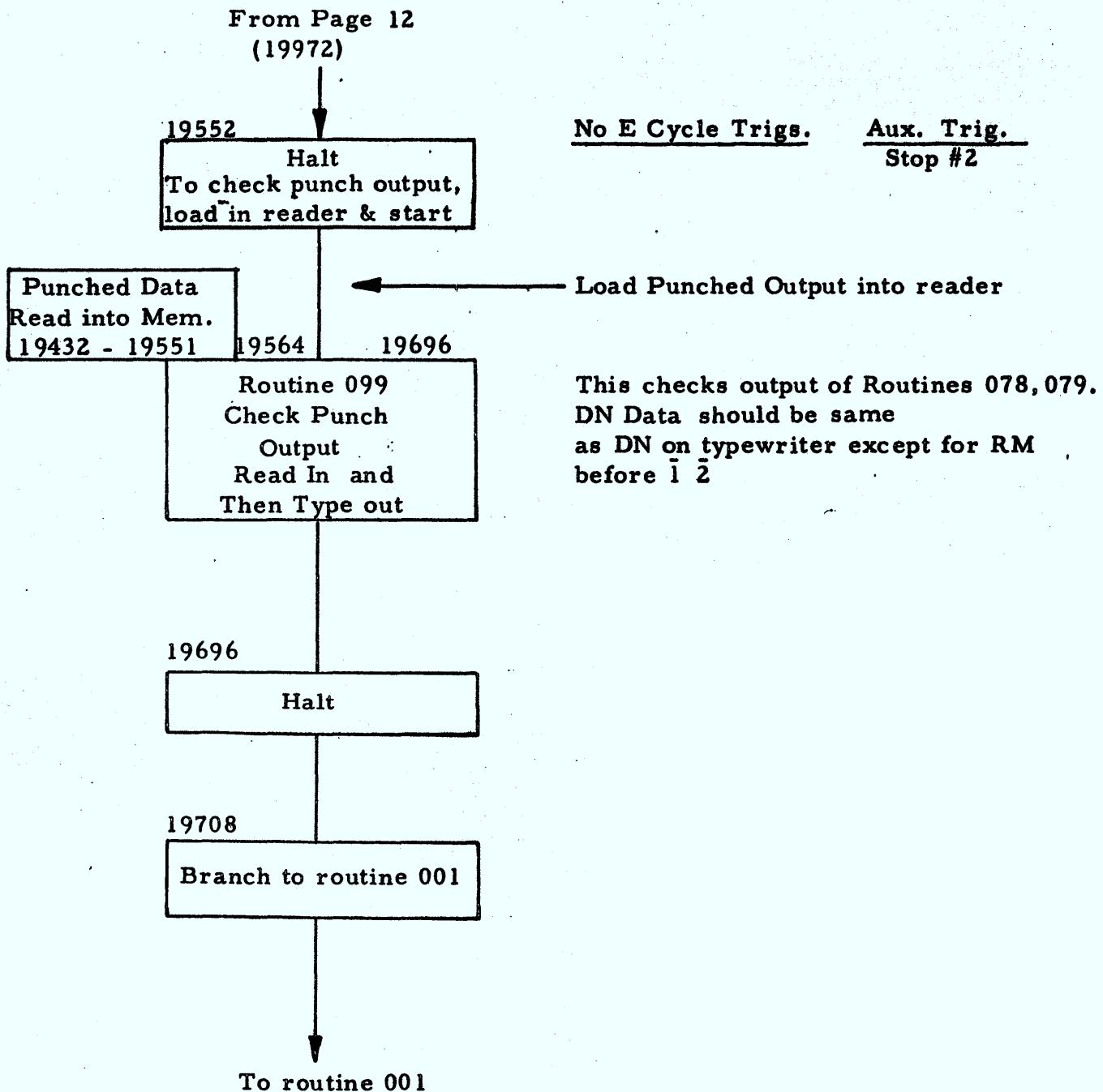
I/O Hold
RSP GATE
R/W Call
Discon. Gate
I/O Exit
I/O Sync.

Branch to
Routine 001



To Page 13
(19552)

PN 2128301
EC 404675



1620 DIAGNOSTIC PROGRAM

CUO1

MEM	00PPPPPQQQQQ	OP
LOC	01 23456 78901	TYP

24		X	
36		X	
48		X	
60		X	
72		X	
84		X	
96	000 00000	MT	MULTIPLY TABLE
108	00 00102 03040	MT	MULTIPLY TABLE
120	00 20406 08000	MT	MULTIPLY TABLE
132	30 60902 10040	MT	MULTIPLY TABLE
144	80 21610 05001	MT	MULTIPLY TABLE
156	51 02006 02181	MT	MULTIPLY TABLE
168	42 00704 11282	MT	MULTIPLY TABLE
180	00 80614 22300	MT	MULTIPLY TABLE
192	90 81726 30000	MT	MULTIPLY TABLE
204	00 00005 06070	MT	MULTIPLY TABLE
216	80 90012 14161	MT	MULTIPLY TABLE
228	81 51811 24272	MT	MULTIPLY TABLE
240	02 42822 36352	MT	MULTIPLY TABLE
252	03 53045 40363	MT	MULTIPLY TABLE
264	24 84455 32494	MT	MULTIPLY TABLE
276	65 36048 46546	MT	MULTIPLY TABLE
288	27 54453 62718	MT	MULTIPLY TABLE
300	01 23456 78912	AT	ADD TABLES
312	34 56789 02345	AT	ADD TABLES
324	67 89013 45678	AT	ADD TABLES
336	90 12456 78901	AT	ADD TABLES
348	23 56789 01234	AT	ADD TABLES
360	67 89012 34578	AT	ADD TABLES
372	90 12345 68901	AT	ADD TABLES
384	23 45679 01234	AT	ADD TABLES
396	56 787	AT	ADD TABLES
408		X	
420		X	
432		X	
444		X	
456		X	
468		X	
480		X	
492	62 63415 963	X	START
504	59 56646 34955	X	ROUTIN
516	45 6203 4563	X	ES. ET
528	56 62 4 65653	X	OS FOL

 PN 2128301
 EC 404568

540	53	56660	3	0#	X	LOW.
552	34		00102		K	CARRIAGE RETURN
564	39	00493	00100		WA	START ROUTINES. ETOS FOLLOW
576	49	01116			B	
588					X	

ROUTINE 100
TYPES SENSE SW SETTINGS

600	62	66	7	1	56	X	SWITCH SETUP DATA
612	55	0#6	266			X	SWITCH SETUP DATA
624	71	564	646			X	SWITCH SETUP DATA
636	0#	6266		72		X	SWITCH SETUP DATA
648	56	55	0	#6266		X	SWITCH SETUP DATA
660		72	5	64646		X	SWITCH SETUP DATA
672		0#626	6	73		X	SWITCH SETUP DATA
684			5655	0#62		X	SWITCH SETUP DATA
696	66	73		5646		X	SWITCH SETUP DATA
708	46	0#6	266			X	SWITCH SETUP DATA
720	74	565	5	0#		X	SWITCH SETUP DATA
732	62	66	7	4	56	X	SWITCH SETUP DATA
744	46	46	0	#6245		X	SWITCH SETUP DATA
756	63	626	662			X	SWITCH SETUP DATA
768	46	5659		43		X	SWITCH SETUP DATA
780	64	70710	3			X	SWITCH SETUP DATA
792		63484	555			X	SWITCH SETUP DATA
804	62	63415	96303			X	SWITCH SETUP DATA
816	00	0#				X	
828	46	00852	00100		BI		CHECK FOR SW 1 ON
840	47	00876	00100		BNI		CHECK FOR SW 1 OFF
852	39	00601	00100		WA		SW 1 ON
864	49	00888			B		
876	39	00619	00100		WA		SW 1 OFF
888	46	00912	00200		BI		CHECK FOR SW 2 ON
900	47	00936	00200		BNI		CHECK FOR SW 2 OFF
912	39	00639	00100		WA		SW 2 ON
924	49	00948			B		
936	39	00657	00100		WA		SW 2 OFF
948	46	00972	00300		BI		CHECK FOR SW 3 ON
960	47	00996	00300		BNI		CHECK FOR SW 3 OFF
972	39	00677	00100		WA		SW 3 ON

984	49 01008	B	
996	39 00695 00100	WA	SW 3 OFF
1008	46 01032 00400	BI	CHECK FOR SW 4 ON
1020	47 01056 00400	BNI	CHECK FOR SW 4 OFF
1032	39 00715 00100	WA	SW 4 ON
1044	49 01068	B	
1056	39 00733 00100	WA	SW 4 OFF
1068	39 00753 00100	WA	SET SWS FOR CU01 THEN START
1080	48	H	
1092	49 00552	B	

ROUTINE 001
BRANCH NO RECORD MARK ON RM EVEN MEMORY PO

1104	#	X	CONSTANTS
1116	45 01152 01114	BNR	CHECK FOR RM
1128	49 01200	B	
1140	41	NOP	
			ERROR ROUTINE
1152	46 01176 00100	BI	CHECK SW 1 IF ON BY ETO
1164	39 01189 00100	WA	
1176	47 01200 00300	BNI	CHECK SW 3 IF ON STOP ERROR
1188	48 70707 1 07	H	
1200	46 01116 00200	BI	CHECK SW 2 IF ON LOOP ROUTINE
1212	49 01236	B	

ROUTINE 002
BRANCH NO RECORD MARK ON RM ODD MEMORY POS

1224	#	X	CONSTANTS
1236	45 01272 01235	BNR	CHECK FOR RM
1248	49 01320	B	
1260	41	NOP	
			ERROR ROUTINE
1272	46 01296 00100	BI	

1284	39 01309	00100	WA
1296	47 01320	00300	BNI
Q1320	48 70707	2 07	H
1320	46 01236	00200	BI
J1332	49 01356		B

ROUTINE 003
BRANCH NO RECORD MARK ON 8 IN EVEN MEMORY POSITION

144	8	X	CONSTANTS	
1356	45 01428	01354	BNR	CHECK FOR NO RM
1368	49 01380		B	ERROR ROUTINE
1380	46 01404	00100	BI	
1392	39 01417	00100	WA	
404	47 01428	00300	BNI	
1416	48 70707	3 07	H	
Q128	46 01356	00200	BI	
1440	49 01464		B	

ROUTINE 004
BRANCH NO RECORD MARK ON 8 IN ODD MEMORY POSITION

1452	8	X	CONSTANTS	
Q1464	45 01536	01463	BNR	CHECK FOR NO RM
1476	49 01488		B	ERROR ROUTINE
Q188	46 01512	00100	BI	
1500	39 01525	00100	WA	
1512	47 01536	00300	BNI	
Q1524	48 70707	4 07	H	
1536	46 01464	00200	BI	
Q1548	49 01572		B	

ROUTINE 005
BRANCH NO RECORD MARK ON 2 IN EVEN MEMORY POS

1560	2	X	CONSTANTS
1572	45 01644 01570	BNR	CHECK FOR NO RM
1584	49 01596	B	
			ERROR ROUTINE
1596	46 01620 00100	BI	
1608	39 01633 00100	WA	
1620	47 01644 00300	BNI	
1632	48 70707 5 0#	H	
1644	46 01572 00200	BI	
1656	49 01680	B	

ROUTINE 006
BRANCH NO RECORD MARK ON 2 IN ODD MEMORY POS

1668	2	X	CONSTANTS
1680	45 01752 01679	BNR	CHECK FOR NO RM
1692	49 01704	B	
			ERROR ROUTINE
1704	46 01728 00100	BI	
1716	39 01741 00100	WA	
1728	47 01752 00300	BNI	
1740	48 70707 6 0#	H	
1752	46 01680 00200	BI	
1764	49 01788	B	

ROUTINE 007
BRANCH NO RECORD MARK ON ZERO IN EVEN MEMORY POS

1776	0	X	CONSTANTS
1788	45 01860 01786	BNR	CHECK FOR NO RM
1800	49 01812	B	

ERROR ROUTINE

1812	46	01836	00100	BI
1824	39	01849	00100	WA
1836	47	01860	00300	BNI
1848	48	70707	7 0#	H
1860	46	01788	00200	BI
1872	49	01896		B

ROUTINE 008**BRANCH NO RECORD MARK ON ZERO IN ODD MEMORY POS**

1884		0	X	CONSTANTS
1896	45	01968	01895	BNR CHECK FOR NO RM
1908	49	01920		B
ERROR ROUTINE				
1920	46	01944	00100	BI
1932	39	01957	00100	WA
1944	47	01968	00300	BNI
1956	48	70707	8 0#	H
1968	46	01896	00200	BI
1980	49	02004		B

ROUTINE 009**BRANCH NO FLAG ON FLAG EVEN MEMORY POS**

1992		I	X	CONSTANTS
2004	44	02028	02002	BNF CHECK FOR FLAG
2016	49	02076		B
ERROR ROUTINE				
2028	46	02052	00100	BI
2040	39	02065	00100	WA
2052	47	02076	00300	BNI
2064	48	70707	9 0#	H
2076	46	02004	00200	BI
2088	49	02112		B

ROUTINE 010
BRANCH NO FLAG ON FLAG ODD MEMORY POS

2100	I	X	CONSTANTS
2112	44 02136	02111	BNF CHECK FOR FLAG
2124	49 02184		B
			ERROR ROUTINE
2136	46 02160	00100	BI
2148	39 02173	00100	WA
2160	47 02184	00300	BNI
2172	48 70717	0 0#	H
2184	46 02112	00200	BI
2196	49 02220		B

ROUTINE 011
BRANCH NO FLAG ON NO FLAG EVEN MEMORY POS

2208	0	X	CONSTANTS
2220	44 02292	02218	BNF
2232	49 02244		B
			ERROR ROUTINE
2244	46 02268	00100	BI
2256	39 02281	00100	WA
2268	47 02292	00300	BNI
2280	48 70717	1 0#	H
2292	46 02220	00200	BI
2304	49 02328		B

ROUTINE 012
BRANCH NO FLAG ON NO FLAG ODD MEMORY POS

2316	0	X	CONSTANTS
2328	44 02400	02325	BNF
2340	49 02352		B

PN 2128301
 EC 404530

ERROR ROUTINE

2352	46 02376 00100	BI
2364	39 02389 00100	WA
2376	47 02400 00300	BNI
2388	48 70717 2 0?	H
2400	46 02328 00200	BI
2412	49 02436	B

**ROUTINE 013
BRANCH ON DIGIT 1**

2424	1	X	CONSTANTS
2436	43 02496 02434	BD	CHECK FOR A 1
			ERROR ROUTINE
2448	46 02472 00100	BI	
2460	39 02485 00100	WA	
2472	47 02496 00300	BNI	
2484	48 70717 3 0?	H	
2496	46 02436 00200	BI	
2508	49 02532	B	

**ROUTINE 014
BRANCH ON DIGIT 2**

2520	2	X	CONSTANTS
2532	43 02592 02531	BD	CHECK FOR A 2
			ERROR ROUTINE
2544	46 02568 00100	BI	
2556	39 02581 00100	WA	
2568	47 02592 00300	BNI	
2580	48 70717 4 0?	H	
2592	46 02532 00200	BI	
2604	49 02628	B	

ROUTINE 015
BRANCH ON DIGIT 4

2616	4	X	CONSTANTS
2628	43 02688 02626	BD	CHECK FOR A 4
			ERROR ROUTINE
2640	46 02664 00100	BI	
2652	39 02677 00100	WA	
2664	47 02688 00300	BNI	
2676	48 70717 5 0?	H	
2688	46 02628 00200	BI	
2700	49 02724	B	

ROUTINE 016
BRANCH ON DIGIT 8

2712	8	X	CONSTANTS
2724	43 02784 02723	BD	CHECK FOR AN 8
			ERROR ROUTINE
2736	46 02760 00100	BI	
2748	39 02773 00100	WA	
2760	47 02784 00300	BNI	
2772	48 70717 6 0?	H	
2784	46 02724 00200	BI	
2796	49 02820	B	

ROUTINE 017
BRANCH ON DIGIT 0

2808	0	X	CONSTANTS
2820	43 02844 02818	BD	CHECK FOR ZERO
2832	49 02892	B	

ERROR ROUTINE

2844	46 02868 00100	BI
2856	39 02881 00100	WA
2868	47 02892 00300	BNI
2880	48 70717 7 0#	H
2892	46 02820 00200	BI
2904	49 02928	B

ROUTINE 018

TRANS DIGIT FROM EVEN TO EVEN MEMORY POS

2916	0 #	X	CONSTANTS AND WORKING AREA
2928	25 02922 02926	TD	TRANS RM
2940	45 03000 02922	BNR	CHECK FOR RM
2952	25 02922 02924	TD	TRANS ZERO
2964	43 03000 02922	BD	CHECK FOR NO DIGIT
2976	49 03048	B	
2988	41	NOP	

ERROR ROUTINE

3000	46 03024 00100	BI
3012	39 03037 00100	WA
3024	47 03048 00300	BNI
3036	48 70717 8 0#	H
3048	46 02928 00200	BI
3060	49 03084	B

ROUTINE 019

TRANS DIGIT FROM ODD TO ODD MEMORY POS

3072	0 #	X	CONSTANTS AND WORKING AREA
3084	25 03077 03083	TD	TRANS RM
3096	45 03156 03077	BNR	CHECK FOR RM
3108	25 03077 03081	TD	TRANS ZERO

3120	43 03156 03077	BD	CHECK FOR NO DIGIT
3132	49 03204	B	
3144	41	NOP	
			ERROR ROUTINE
3156	46 03180 00100	BI	
3168	39 03193 00100	WA	
3180	47 03204 00300	BNI	
3192	48 70717 9 0#	H	
3204	46 03084 00200	BI	
3216	49 03240	B	

ROUTINE 020
TRANS DIGIT FROM EVEN TO ODD MEMORY POSITION

3228	0 #	X	CONSTANTS AND WORKING AREA
3240	25 03233 03238	TD	TRANS RM
3252	45 03312 03233	BNR	CHECK FOR RM
3264	25 03233 03236	TD	TRANS ZERO
3276	43 03312 03233	BD	CHECK FOR NO DIGIT
3288	49 03360	B	
3300	41	NOP	
			ERROR ROUTINE
3312	46 03336 00100	BI	
3324	39 03349 00100	WA	
3336	47 03360 00300	BNI	
3348	48 70727 0 0#	H	
3360	46 03240 00200	BI	
3372	49 03396	B	

ROUTINE 021
TRANS DIGIT FROM ODD TO EVEN MEMORY POS

3384	0 #	X	CONSTANTS AND WORKING AREA
3396	25 03390 03395	TD	TRANS RM
3408	45 03468 03390	BNR	CHECK FOR RM

3420	25 03390 03393	TD	TRANS ZERO
3432	43 03468 03390	BD	CHECK FOR NO DIGIT
3444	49 03516	B	
3456	41	NOP	
			ERROR ROUTINE
3468	46 03492 00100	BI	
3480	39 03505 00100	WA	
3492	47 03516 00300	BNI	
3504	48 70727 1 0#	H	
3516	46 03396 00200	BI	
3528	49 03552	B	

ROUTINE 022
TRANS IMMED RECORD MARK TO EVEN MEMORY POS

3540	X	WORKING AREA	
3552	15 03546 0000#	TDM	TRANS IMMED RM
3564	45 03624 03546	BNR	CHECK FOR RM
3576	15 03546 00000	TDM	TRANS IMMED ZERO
3588	43 03624 03546	BD	CHECK FOR NO DIGIT
3600	49 03672	B	
3612	41	NOP	
			ERROR ROUTINE
3624	46 03648 00100	BI	
3636	39 03661 00100	WA	
3648	47 03672 00300	BNI	
3660	48 70727 2 0#	H	
3672	46 03552 00200	BI	
3684	49 03708	B	

ROUTINE 023
TRANS IMMED RECORD MARK TO ODD MEMORY POS

3696		X	WORKING AREA
3708	15 03701 0000#	TDM	TRANS IMMED RM
3720	45 03780 03701	BNR	CHECK FOR RM
3732	15 03701 00000	TDM	TRANS IMMED ZERO
3744	43 03780 03701	BD	CHECK FOR NO DIGIT
3756	49 03828	B	
3768	41	NOP	
ERROR ROUTINE			
3780	46 03804 00100	BI	
3792	39 03817 00100	WA	
3804	47 03828 00300	BNI	
3816	48 70727 3 0#	H	
3828	46 03708 00200	BI	
3840	49 03864	B	

ROUTINE 024
TRANS FIELD-2 CHAR (1 #) TO ODD MEMORY POS

3852	17	X	CONSTANTS AND WORKING AREA
3864	26 03857 03863	TF	TRANS 1#
3876	45 03924 03857	BNR	CHECK FOR RM
3888	44 03924 03856	BNF	CHECK FOR FLAG
3900	49 03972	B	
3912	41	NOP	
ERROR ROUTINE			
3924	46 03948 00100	BI	
3936	39 03961 00100	WA	
3948	47 03972 C0300	BNI	
3960	48 70727 4 0#	H	
3972	46 03864 00200	BI	
3984	49 04008	B	

PN 2128301
 EC 404675

ROUTINE 025
TRANS FIELD - 2 CHAR (I#) TO EVEN MEMORY POS

3996	I#	X	CONSTANTS AND WORKING AREA
4008	26 04002 04006	TF	TRANS I#
4020	45 04068 04002	BNR	CHECK FOR RM
4032	44 04068 04001	BNF	CHECK FOR FLAG
4044	49 04116	B	
4056	41	NOP	
			ERROR ROUTINE
4068	46 04092 00100	BI	
4080	39 04105 00100	WA	
4092	47 04116 00300	BNI	
4104	48 70727 5 0#	H	
4116	46 04008 00200	BI	
4128	49 04152	B	

ROUTINE 026
TRANS FIELD IMMED - 3 CHAR (I7#)

4140	X	WORKING AREA	
4152	16 04145 00I7#	TFM	TRANS I7#
4164	45 04212 04145	BNR	CHECK FOR RM
4176	44 04212 04143	BNF	CHECK FOR FLAG
4188	49 04260	B	
4200	41	NOP	
			ERROR ROUTINE
4212	46 04236 00100	BI	
4224	39 04249 00100	WA	
4236	47 04260 00300	BNI	
4248	48 70727 6 0#	H	
4260	46 04152 00200	BI	
4272	49 04308	B	

ROUTINE 027
TRANS RECORD-6 CHAR (I24807) TO ODD MEMORY POS

4284	I 24807	X	CONSTANTS
4296		X	WORKING AREA
4308	31 04301 04290	TR	TRANS RECORD - I24807
4320	44 04368 04301	BNF	CHECK FOR FIRST CHARACTER
4332	45 04368 04306	BNR	CHECK FOR LAST CHARACTER
4344	49 04416	B	
4356	41	NOP	
			ERROR ROUTINE
4368	46 04392 00100	BI	
4380	39 04405 00100	WA	
4392	47 04416 00300	BNI	
4404	48 70727 7 0#	H	
4416	46 04308 00200	BI	
4428	49 04452	B	

ROUTINE 028
BRANCH AND TRANS 6 CHAR (I24807)

4440	I 24807	X	CONSTANTS
4452	27 04596 04451	BT	BRANCH TO 04596 AND TRANS FIELD
4464	44 04656 04594	BNF	CHECK 04594 FOR FLAG
4476	49 04704	B	
4488	41	NOP	
4500	41	NOP	
			ERROR ROUTINE
4512	46 04536 00100	BI	
4524	39 04549 00100	WA	
4536	47 04560 00300	BNI	
4548	48 70727 8 0#	H	
4560	46 04452 00200	BI	
4572	49 04620	B	

SUB-ROUTINE 528

THIS IS ROUTINE BRANCHED TO IN 028. CHKS TRANS CORRECT. CHANGES 0 TO I and BB to MAIN ROUTINE

4584		X	WORKING AREA
4596	45 04512 04595	BNR	CHECK LOW ORDER FOR RM
4608	44 04512 04590	BNF	CHECK HIGH ORDER FOR FLAG
4620	15 04594 00001	TDM	TRANS I TO 04594

4632	16 04593	00000	TFM	CLEAR TRANSMITTED FIELD
4644	42 04692		BB	BRANCH BACK TO 04464
				ERROR ROUTINE
4656	46 04680	00100	BI	
4668	39 04693	00100	WA	
4680	47 04704	00300	BNI	
4692	48 75727	8 07	H	
4704	46 04452	00200	BI	
4716	49 04728		B	

ROUTINE 029
BRANCH AND TRANS IMMED TRANS 3 CHAR FIELD (T7)

4728	17 04872	00177	BTM	BRANCH TO 04872 AND TRANS FIELD
4740	44 04932	04868	BNF	CHECK 04868 FOR FLAG
4752	49 04980		B	
4764	41		NOP	
4776	41		NOP	
				ERROR ROUTINE
4788	46 04812	00100	BI	
4800	39 04825	00100	WA	
4812	47 04836	00300	BNI	
4824	48 70727	9 07	H	
4836	46 04728	00200	BI	
4848	49 04896		B	

SUB-ROUTINE 529
THIS IS ROUTINE BRANCHED TO IN 029. CHKS TRANS
CORRECT. CHANGES 7 TO 1 AND BB TO MAIN ROUTINE

4860	X	WORKING AREA
4872	BNR	CHECK LOW ORDER FOR RM
4884	BNF	CHECK HIGH ORDER FOR FLAG
4896	TFM	TRANS FIELD T01 IMMED
4908	BB	BRANCH BACK TO 04740
4920	NOP	

ERROR ROUTINE

4932	46 04956 00100	BI
4944	39 04969 00100	WA
4956	47 04980 00300	BNI
4968	48 75727 9 07	H
4980	46 04728 00200	BI
4992	49 05016	B

ROUTINE 030

SET FLAG ON CHAR WITH FLAG AND C BIT (8)

5004	8	X	WORKING AREA
5016	46 05028 01600	BI	TURN OFF MBR E CHECK
5028	32 05014	SF	SET FLAG ON 8
5040	44 05076 05014	BNF	CHECK FLAG NOT REMOVED
5052	46 05076 01600	BI	CHECK C BIT NOT REMOVED
5064	49 05124	B	

ERROR ROUTINE

5076	46 05100 00100	BI
5088	39 05113 00100	WA
5100	47 05124 00300	BNI
5112	48 70737 0 07	H
5124	46 05016 00200	BI
5136	49 05160	B

ROUTINE 031

SET FLAG ON CHAR WITH FLAG AND NO C BIT 6

5148	6	X	WORKING AREA
5160	46 05172 01700	BI	TURN OFF MBR O CHECK
5172	32 05157	SF	SET FLAG ON 6
5184	44 05220 05157	BNF	CHECK FLAG NOT REMOVED
5196	46 05220 01700	BI	CHECK C BIT NOT REMOVED
5208	49 05268	B	

ERROR ROUTINE

5220	46 05244 00100	BI
5232	39 05257 00100	WA
5244	47 05268 00300	BNI
5256	48 70737 1 0\$	H
5268	46 05160 00200	BI
5280	49 05304	B

ROUTINE 032

SET FLAG ON CHAR WITH C BIT AND NO FLAG(5)

5292	5	X	WORKING AREA
5304	32 05302	SF	SET FLAG ON 5
5316	44 05376 05302	BNF	CHECK FOR FLAG
5328	46 05376 01600	BI	CHECK C BIT REMOVED
5340	15 05302 00005	TDM	RESTORE TO 5
5352	49 05424	B	
5364	41	NOP	

ERROR ROUTINE

5376	46 05400 00100	BI
5388	39 05413 00100	WA
5400	47 05424 00300	BNI
5412	48 70737 2 0\$	H
5424	46 05304 00200	BI
5436	49 05460	B

ROUTINE 033

SET FLAG BIT ON CHAR WITH NO C BIT OR FLAG(4)

5448	4	X	WORKING AREA
5460	32 05459	SF	SET FLAG ON 4
5472	44 05532 05459	BNF	CHECK FOR FLAG
5484	46 05532 01700	BI	CHECK C BIT

5496	15 05459 00004	TDM	RESTORE TO 4
5508	49 05580	B	
5520	41	NOP	
			ERROR ROUTINE
5532	46 05556 00100	BI	
5544	39 05569 00100	WA	
5556	47 05580 00300	BNI	
5568	48 70737 3 0#	H	
5580	46 05460 00200	BI	
5592	49 05616	B	

ROUTINE 034
CLEAR FLAG ON CHAR WITH NO FLAG OR C BIT(1)

5604	1	X	WORKING AREA
5616	33 05614	CF	CLEAR FLAG ON 1
5628	44 05652 05614	BNF	CHECK FLAG NOT ADDED
5640	49 05688	B	ENTER ERROR IF FLAG
5652	46 05688 01600	BI	CHECK C BIT NOT INSERTED
5664	49 05736	B	
5676	41	NOP	
			ERROR ROUTINE
5688	46 05712 00100	BI	
5700	39 05725 00100	WA	
5712	47 05736 00300	BNI	
5724	48 70737 4 0#	H	
5736	46 05616 00200	BI	
5748	49 05772	B	

ROUTINE 035
CLEAR FLAG ON CHAR WITH NO FLAG BUT WITH C BIT(3)

5760	3	X	WORKING AREA
5772	33 05769	CF	CLEAR FLAG
5784	44 05808 05769	BNF	CHECK FLAG NOT INSERTED

5796	49 05844	B	ENTER ERROR IF FLAG
5808	46 05844 01700	BI	CHECK C BIT NOT REMOVED
5820	49 05892	B	
5832	41	NOP	
			ERROR ROUTINE
5844	46 05868 00100	BI	
5856	39 05881 00100	WA	
5868	47 05892 00300	BNI	
5880	48 70737 5 0#	H	
5892	46 05772 00200	BI	
5904	49 05928	B	

ROUTINE 036
CLEAR FLAG ON CHAR WITH FLAG BUT NO C BIT (5)

5916	5	X	WORKING AREA
5928	33 05926	CF	CLEAR FLAG ON 5
5940	44 05964 05926	BNF	CHECK FLAG REMOVED
5952	49 06000	B	ENTER ERROR IF FLAG
5964	46 06000 01600	BI	CHECK IF C BIT INSERTED
5976	32 05926	SF	RESTORE FLAG
5988	49 06048	B	
			ERROR ROUTINE
6000	46 06024 00100	BI	
6012	39 06037 00100	WA	
6024	47 06048 00300	BNI	
6036	48 70737 6 0#	H	
6048	46 05928 00200	BI	
6060	49 06084	B	

ROUTINE 037
CLEAR FLAG ON CHAR WITH FLAG AND C BIT (7)

6072	7	X	WORKING AREA
6084	33 06081	CF	CLEAR FLAG ON 7
6096	44 06120 06081	BNF	CHECK FOR NO FLAG

6108	49 06156	B	ENTER ERROR IF FLAG
6120	46 06156 01700	BI	CHECK C BIT REMOVED
6132	32 06081	SF	RESTORE FLAG
6144	49 06204	B	
ERROR ROUTINE			
6156	46 06180 00100	BI	
6168	39 06193 00100	WA	
6180	47 06204 00300	BNI	
6192	48 70737 7 07	H	
6204	46 06084 00200	BI	
6216	49 06240	B	

ROUTINE 038
CHECK HI-POS AND EQ-ZERO TRIGS FOR H/P NOT E/Z

6228	11	X	CONSTANTS AND WORKING AREA
6240	26 06233 06239	TF	SET ONES IN P FIELD
6252	21 06233 06239	A	ADD 11 TO 11. RESULTS H/P
6264	47 06324 01100	BNI	CHECK H/P TRIG FOR H/P
6276	46 06360 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
6288	49 06408	B	
6300	41	NOP	
6312	41	NOP	
ERROR ROUTINE			
6324	46 06276 00100	BI	
6336	38 06355 00100	WN	
6348	49 06276 0387	B	
6360	46 06384 00100	BI	
6372	39 06397 00100	WA	
6384	47 06408 00300	BNI	
6396	48 75737 8 07	H	
6408	46 06240 00200	BI	
6420	49 06444	B	

ROUTINE 039
CHK HI-POS AND EQ-ZERO TRIGS FOR NOT H/P NOT E/Z

6432	22	X	CONSTANTS AND WORKING AREA
6444	26 06437 06442	TF	SET MINUS 22 IN P FIELD
6456	21 06437 06442	A	ADD -22 TO -22
6468	46 06528 01100	BI	CHECK H/P TRIG FOR NOT H/P
6480	46 06564 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
6492	49 06612	B	
6504	41	NOP	
6516	41	NOP	
			ERROR ROUTINE
6528	46 06480 00100	BI	
6540	38 06559 00100	WN	
6552	49 06480 039#	B	
6564	46 06588 00100	BI	
6576	39 06601 00100	WA	
6588	47 06612 00300	BNI	
6600	48 75737 9 0#	H	
6612	46 06444 00200	BI	
6624	49 06648	B	

ROUTINE 040
CHK HI-POS AND EQ-ZERO TRIGS FOR NOT H/P NOT E/Z

6636	8811	X	CONSTANTS AND WORKING AREA
6648	26 06641 06647	TF	SET 11 IN P FIELD
6660	21 06641 06645	A	ADD MINUS 88 TO 11
6672	46 06732 01100	BI	CHECK H/P FOR NOT H/P
6684	46 06768 01200	BI	CHECK E/Z FOR NOT E/Z
6696	49 06816	B	
6708	41	NOP	
6720	41	NOP	
			ERROR ROUTINE
6732	46 06684 00100	BI	
6744	38 06763 00100	WN	
6756	49 06684 040#	B	
6768	46 06792 00100	BI	
6780	39 06805 00100	WA	
6792	47 06816 00300	BNI	
6804	48 75747 0 0#	H	
6816	46 06648 00200	BI	
6828	49 06852	B	

ROUTINE 041
CHK HI-POS AND EQ-ZERO TRIGS FOR H/P NOT E/Z

6840	8844	X	CONSTANTS AND WORKING AREA
6852	26 06845 06850	TF	SET MINUS 44 IN P FIELD
6864	21 06845 06848	A	ADD 88 TO -44
6876	47 06936 01100	BNI	CHECK H/P FOR H/P
6888	46 06972 01200	BI	CHECK E/Z FOR NOT E/Z
6900	49 07020	B	
6912	41	NOP	
6924	41	NOP	
ERROR ROUTINE			
6936	46 06888 00100	BI	
6948	38 06967 00100	WN	
6960	49 06888 041#	B	
6972	46 06996 00100	BI	
6984	39 07009 00100	WA	
6996	47 07020 00300	BNI	
7008	48 75747 1 0#	H	
7020	46 06852 00200	BI	
7032	49 07056	B	

ROUTINE 042
CHK HI-POS AND EQ-ZERO TRIGS FOR E/Z NOT H/P

7044	4444	X	CONSTANTS AND WORKING AREA
7056	26 07049 07055	TF	SET 44 IN P FIELD
7068	21 07049 07053	A	ADD MINUS 44 TO 44
7080	46 07140 01100	BI	CHECK H/P TRIG FOR NOT H/P
7092	47 07176 01200	BNI	CHECK E/Z TRIG FOR E/Z
7104	49 07224	B	
7116	41	NOP	
7128	41	NOP	
ERROR ROUTINE			
7140	46 07092 00100	BI	
7152	38 07171 00100	WN	
7164	49 07092 042#	B	
7176	46 07200 00100	BI	
7188	39 07213 00100	WA	
7200	47 07224 00300	BNI	
7212	48 75747 2 0#	H	
7224	46 07056 00200	BI	
7236	49 07260	B	

ROUTINE 043
CHK HI-POS AND EQ-ZERO TRIGS FOR H/P NOT E/Z

7248	1188	X	CONSTANTS AND WORKING AREA
7260	26 07253 07259	TF	SET 88 IN P FIELD
7272	22 07253 07257	S	SUBT 11 FROM 88
7284	47 07344 01100	BNI	CHECK H/P TRIG FOR H/P
7296	46 07380 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
7308	49 07428	B	
7320	41	NOP	
7332	41	NOP	
			ERROR ROUTINE
7344	46 07296 00100	BI	
7356	38 07375 00100	WN	
7368	49 07296 0437	B	
7380	46 07404 00100	BI	
7392	39 07417 00100	WA	
7404	47 07428 00300	BNI	
7416	48 75747 3 07	H	
7428	46 07260 00200	BI	
7440	49 07464	B	

ROUTINE 044
CHK HI-POS AND EQ-ZERO TRIGS FOR NOT H/P NOT E/Z

7452	1188	X	CONSTANTS AND WORKING AREA
7464	26 07457 07461	TF	SET 11 IN P FIELD
7476	22 07457 07463	S	SUBT 88 FROM 11
7488	46 07548 01100	BI	CHECK H/P TRIG FOR NOT H/P
7500	46 07584 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
7512	49 07632	B	
7524	41	NOP	
7536	41	NOP	
			ERROR ROUTINE
7548	46 07500 00100	BI	
7560	38 07579 00100	WN	
7572	49 07500 0447	B	
7584	46 07608 00100	BI	
7596	39 07621 00100	WA	
7608	47 07632 00300	BNI	
7620	48 75747 4 07	H	
7632	46 07464 00200	BI	
7644	49 07668	B	

ROUTINE 045
CHECK HI-POS AND EQ-ZERO TRIGS FOR E/Z, NOT H/P

7656	88	X	CONSTANTS AND WORKING AREA
7668	26 07661 07667	TF	SET 88 IN P FIELD
7680	22 07661 07667	S	SUBT 88 FROM 88
7692	46 07752 01100	BI	CHECK H/P TRIG FOR NOT H/P
7704	47 07812 01200	BNI	CHECK E/Z TRIG FOR E/Z
7716	49 07860	B	
7728	41	NOP	
7740	41	NOP	
ERROR ROUTINE			
7752	46 07776 00100	BI	
7764	39 07789 00100	WA	
7776	47 07800 00300	BNI	
7788	48 70747 5 07	H	
7800	49 07704	B	
7812	46 07836 00100	BI	
7824	39 07849 00100	WA	
7836	47 07860 00300	BNI	
7848	48 75747 5 07	H	
7860	46 07668 00200	BI	
7872	49 07896	B	

ROUTINE 046
CHECK HI-POS AND EQ-ZERO TRIGS FOR H/P NOT E/Z

7884	4422	X	CONSTANTS AND WORKING AREA
7896	26 07889 07895	TF	SET 22 IN P FIELD
7908	22 07889 07893	S	SUBT-44 FROM 22
7920	47 07980 01100	BNI	CHECK H/P TRIG FOR H/P
7932	46 08040 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
7944	49 08088	B	
7956	41	NOP	
7968	41	NOP	
ERROR ROUTINE			
7980	46 08004 00100	BI	
7992	39 08017 00100	WA	
8004	47 08028 00300	BNI	
8016	48 70747 6 07	H	
8028	49 07932	B	

8040	46 08064 00100	BI
8052	39 08077 00100	WA
8064	47 08088 00300	BNI
8076	48 75747 6 07	H
8088	46 07896 00200	BI
8100	49 08124	B

ROUTINE 047
CHK HI-POS AND EQ-ZERO TRIGS FOR NOT H/P NOT E/Z

8112	4422	X	CONSTANTS AND WORKING AREA
8124	26 08117 08122	TF	SET MINUS 22 IN P FIELD
8136	22 08117 08120	S	SUBT 44 FROM - 22
8148	46 08208 01100	BI	CHECK H/P TRIG FOR NOT H/P
8160	46 08268 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
8172	49 08316	B	
8184	41	NOP	
8196	41	NOP	
			ERROR ROUTINE
8208	46 08222 00100	BI	
8220	39 08245 00100	WA	
8232	47 08256 00300	BNI	
8244	48 70747 7 07	H	
8256	49 08160	B	
8268	46 08292 00100	BI	
8280	39 08305 00100	WA	
8292	47 08316 00300	BNI	
8304	48 75747 7 07	H	
8316	46 08124 00200	BI	
8328	49 08352		

ROUTINE 048
CHK HI-POS AND EQ-ZERO TRIGS FOR H/P NOT E/Z

8340	8822	X	CONSTANTS AND WORKING AREA
8352	26 08345 08351	TF	SET -22 IN P FIELD
8364	22 08345 08349	S	SUBT -88 FROM -22
8376	47 08436 01100	BNI	CHECK H/P TRIG FOR H/P

8388	46 08496 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
8400	49 08544	B	
8412	41	NOP	
8424	41	NOP	

ERROR ROUTINE

8436	46 08460 00100	BI
8448	39 08473 00100	WA
8460	47 08484 00300	BNI
8472	48 70747 8 07	H
8484	49 08388	B
8496	46 08520 00100	BI
8508	39 08533 00100	WA
8520	47 08544 00300	BNI
8532	48 75747 8 07	H
8544	46 08352 00200	BI
8556	49 08580	B

RQUTINE 049
CHK HI-POS AND EQ-ZERO TRIGS FOR NOT H/P NOT E/Z

8568	1144	X	CONSTANTS AND WORKING AREA
8580	26 08573 08578	TF	SET -44 IN P FIELD
8592	22 08573 08576	S	SUBT -11 FROM -44
8604	46 08664 01100	BI	CHECK H/P TRIG FOR NOT H/P
8616	46 08724 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
8628	49 08772	B	
8640	41	NOP	
8652	41	NOP	

ERROR ROUTINE

8664	46 08688 00100	BI
8676	39 08701 00100	WA
8688	47 08712 00300	BNI
8700	48 70747 9 07	H
8712	49 08616	B
8724	46 08748 00100	BI
8736	39 08761 00100	WA
8748	47 08772 00300	BNI
8760	48 75747 9 07	H
8772	46 08580 00200	BI
8784	49 08808	B

PN 2128301
EC 404530

ROUTINE 050
CHK H1-POS AND EQ-ZERO TRIGS FOR E/Z NOT H/P

8796	33	X	CONSTANTS AND WORKING AREA
8808	26 08801 08806	TF	SET -33 IN P FIELD
8820	22 08801 08806	S	SUBT -33 FROM -33
8832	46 08892 01100	BI	CHECK H/P TRIG FOR NOT H/P
8844	47 08952 01200	BNI	CHECK E/Z TRIG FOR E/Z
8856	49 09000	B	
8868	41	NOP	
8880	41	NOP	

ERROR ROUTINE

8892	46 08916 00100	BI	
8904	39 08929 00100	WA	
8916	47 08940 00300	BNI	
8928	48 70757 0 0#	H	
8940	49 08844	B	
8952	46 08976 00100	BI	
8964	39 08989 00100	WA	
8976	47 09000 00300	BNI	
8988	48 75757 0 0#	H	
9000	46 08808 00200	BI	
9012	49 09036	B	

ROUTINE 051
CHECK FOR CORRECT MEMORY LOOKUP ON ADD

9024	9966	X	CONSTANTS AND WORKING AREA
9036	15 00369 #	TDM	SET RM IN MEMORY POS 369
9048	26 09029 09035	TF	SET 66 IN P FIELD
9060	21 09029 09033	A	ADD 99 TO 66
9072	45 09120 09029	BNR	CHECK RESULT FOR RM
9084	49 09168	B	
9096	41	NOP	
9108	41	NOP	

ERROR ROUTINE

9120	46 09144 00100	BI	
9132	39 09157 00100	WA	
9144	47 09168 00300	BNI	
9156	48 70757 1 0#	H	
9168	15 00369 5	TDM	RESTORE ADD TABLE POS. 369
9180	46 09036 00200	BI	
9192	49 09216	B	

ROUTINE 052
CHECK FOR CORRECT MEMORY LOOK UP ON ADD

9204	9966	X	CONSTANTS AND WORKING AREA
9216	15 00396	#	TDM SET RM IN MEMORY POS. 396
9228	26 09209	09213	TF SET 99 IN P FIELD
9240	21 09209	09215	A ADD 66 TO 99
9252	45 09300	09209	BNR CHECK RESULT FOR RM
9264	49 09348		B
9276	41		NOP
9288	41		NOP

ERROR ROUTINE

9300	46 09324	00100	B I
9312	39 09337	00100	WA
9324	47 09348	00300	BN I
9336	48 70757	2 0#	H
9348	15 00396	5	TDM RESTORE ADD TABLE POS. 396
9360	46 09216	00200	B I
9372	49 09396		B

ROUTINE 053
CHECK FOR CORRECT MEMORY LOOK UP ON SUBT

9384	9966	X	CONSTANTS AND WORKING AREA
9396	15 00303	#	TDM SET RM IN MEMORY POSITION 303
9408	26 09389	09395	TF SET -66 IN P FIELD
9420	22 09389	09393	S SUBT -99 FROM -66
9432	45 09480	09389	BNR CHECK RESULT FOR RM
9444	49 09528		B
9456	41		NOP
9468	41		NOP

ERROR ROUTINE

9480	46 09504	00100	B I
9492	39 09517	00100	WA
9504	47 09528	00300	BN I
9516	48 70757	3 0#	H
9528	15 00303	3	TDM RESTORE ADD TABLE POS. 303
9540	46 09396	00200	B I
9552	49 09576		B

ROUTINE 054
CHECK FOR CORRECT MEMORY LOOK UP ON SUBT

9564	T	11666	X	CONSTANTS AND WORKING AREA
9576	15	00305	≠	TDM SET RM IN MEMORY POS. 305
9588	26	09569	09572	TF SET 111 IN P FIELD
9600	22	09569	09575	S SUBT 666 FROM 111
9612	45	09660	09568	BNR CHECK RESULT FOR RECORD MARK
9624	49	09708		B
9636	41			NOP
9648	41			NOP
ERROR ROUTINE				
9660	46	09684	00100	B I
9672	39	09697	00100	WA
9684	47	09708	00300	BN I
9696	48	70757	4 0≠	H
9708	15	00305	5	TDM RESTORE ADD TABLE POS. 305
9720	46	09576	00200	B I
9732	49	09756		B

ROUTINE 055
CHECK FOR CORRECT MEMORY LOOK UP ON SUBT

9744	9	99888	X	CONSTANTS AND WORKING AREA
9756	15	00301	≠	TDM SET RM IN MEMORY POS. 301
9768	26	09749	09755	TF SET 888 IN P FIELD
9780	22	09749	09752	S SUBT 999 FROM 888
9792	45	09840	09748	BNR CHECK RESULT FOR RECORD MARK
9804	49	09888		B
9816	41			NOP
9828	41			NOP
ERROR ROUTINE				
9840	46	09864	00100	B I
9852	39	09877	00100	WA
9864	47	09888	00300	BN I
9876	48	70757	5 0≠	H
9888	15	00301	1	TDM RESTORE ADD TABLE POS. 301
9900	46	09756	00200	B I
9912	49	09936		B

ROUTINE 056
CHECK OVERFLOW TRIG

9924	1288	X	CONSTANTS AND WORKING AREA
9936	46 09948 01400	BI	TURN OFF OVERFLOW
9948	26 09929 09935	TF	SET 88 IN P FIELD
9960	21 09929 09933	A	ADD 12 TO 88
9972	47 10020 01400	BNI	CHECK FOR OVERFLOW
9984	49 10068	B	
9996	41	NOP	
10008	41	NOP	ERROR ROUTINE
10020	46 10044 00100	BI	
10032	39 10057 00100	WA	
10044	47 10068 00300	BNI	
10056	48 70757 6 0#	H	
10068	46 09936 00200	BI	
10080	49 10104	B	

ROUTINE 057
CHECK OVERFLOW TRIG

10092	55	X	CONSTANTS AND WORKING AREA
10104	46 10116 01400	BI	TURN OFF OVERFLOW
10116	26 10097 10103	TF	SET 55 IN P FIELD
10128	11 10097 222	AM	ADD 222 TO 55 IMMED
10140	47 10188 01400	BNI	CHECK FOR OVERFLOW
10152	49 10236	B	
10164	41	NOP	
10176	41	NOP	ERROR ROUTINE
10188	46 10212 C0100	BI	
10200	39 10225 00100	WA	
10212	47 10236 00300	BNI	
10224	48 70757 7 0#	H	
10236	46 10104 00200	BI	
10248	49 10272	B	

PN 2128301
EC 404530

ROUTINE 058
CHECK OVERFLOW TRIG

10260		73	X	CONSTANTS AND WORKING AREA
10272	46	10284 01400	BI	TURN OFF OVERFLOW
10284	26	10265 10271	TF	SET -73 IN P FIELD
10296	12	10265 27	SM	SUBT 27 FROM -73 IMMED
10308	47	10356 01400	BNI	CHECK FOR OVERFLOW
10320	49	10404	B	
10332	41		NOP	
10344	41		NOP	
				ERROR ROUTINE
10356	46	10380 00100	BI	
10368	39	10393 00100	WA	
10390	47	10404 00300	BNI	
10392	48	70757 8 0#	H	
10404	46	10272 00200	BI	
10416	49	10440	B	

ROUTINE 059
CHECK OVERFLOW TRIG

10428		99	X	CONSTANTS AND WORKING AREA
10440	46	10452 01400	BI	TURN OFF OVERFLOW
10452	26	10433 10439	TF	SET 99 IN P FIELD
10464	12	10433 111	SM	SUBT 111 FROM 99
10476	47	10524 01400	BNI	CHECK FOR OVERFLOW
10488	49	10572	B	
10500	41		NOP	
10512	41		NOP	
				ERROR ROUTINE
10524	46	10548 00100	BI	
10536	39	10561 00100	WA	
10548	47	10572 00300	BNI	
10560	48	70757 9 0#	H	
10572	46	10440 00200	BI	
10584	49	10608	B	

ROUTINE 060
CHECK COMPARE FOR H/P

10596	4488	XX	CONSTANTS AND WORKING AREA
10608	26 10601 10607	TF	SET 88 IN P FIELD
10620	24 10601 10605	C	COMPARE 44 TO 88 RESULT H/P
10632	47 10680 01100	BNI	CHECK H/P TRIG FOR H/P
10644	49 10728	B	
10656	41	NOP	
10668	41	NOP	

ERROR ROUTINE

10680	46 10704 00100	BI	
10692	39 10717 00100	WA	
10704	47 10728 00300	BNI	
10716	48 70767 0 0/	H	
10728	46 10608 00200	BI	
10740	49 10764	B	

ROUTINE 061
CHECK COMPARE FOR H/P

10752	9911	X	CONSTANTS AND WORKING AREA
10764	26 10757 10763	TF	SET 11 IN P FIELD
10776	24 10757 10761	C	COMPARE -22 TO 11
10788	47 10836 01100	BNI	CHECK H/P TRIG FOR H/P
10800	46 10836 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
10812	46 10836 01400	BI	CHECK O/F TRIG FOR NO O/F
10824	49 10884	B	

ERROR ROUTINE

10836	46 10860 00100	BI	
10848	39 10873 00100	WA	
10860	47 10884 00300	BNI	
10872	48 70767 1 0/	H	
10884	46 10764 00200	BI	
10896	49 10920	B	

ROUTINE 062
CHECK COMPARE FOR H/P

10908	7958	X	CONSTANTS AND WORKING AREA
10920	26 10913 10919	TF	SET -58 IN P FIELD
10932	24 10913 10917	C	COMPARE -79 TO -58
10944	47 10992 01100	BNI	CHECK H/P TRIG FOR H/P
10956	49 11040	B	
10968	41	NOP	
10980	41	NOP	
ERROR ROUTING			
10992	46 11016 00100	BI	
11004	39 11029 00100	WA	
11016	47 11040 00300	BNI	
11028	48 70767 2 07	H	
11040	46 10920 00200	BI	
11052	49 11076	B	

ROUTINE 063
CHECK COMPARE FOR E/Z

11064	79	X	CONSTANTS AND WORKING AREA
11076	26 11069 11075	TF	SET 79 IN P FIELD
11088	24 11069 11075	C	COMPARE 79 TO 79
11100	47 11148 01200	BNI	CHECK E/Z TRIG FOR E/Z
11112	49 11196	B	
11124	41	NOP	
11136	41	NOP	
ERROR ROUTINE			
11148	46 11172 00100	BI	
11160	39 11185 00100	WA	
11172	47 11196 00300	BNI	
11184	48 70767 3 07	H	
11196	46 11076 00200	BI	
11208	49 11232	B	

ROUTINE 064
CHECK COMPARE FOR E/Z

11220	68	X	CONSTANTS AND WORKING AREA
11232	26 11225 11230	TF	SET -68 IN P FIELD
11244	24 11225 11230	C	COMPARE -68 TO -68
11256	47 11304 01200	BNI	CHECK E/Z TRIG FOR E/Z
11268	49 11352	B	
11280	41	NOP	
11292	41	NOP	
ERROR ROUTINE			
11304	46 11328 00100	BI	
11316	39 11341 00100	WA	
11328	47 11352 00300	BNI	
11340	48 70767 4 07	H	
11352	46 11232 00200	BI	
11364	49 11388	B	

ROUTINE 065
CHECK COMPARE FOR NOT H/P NOT E/Z

11376	8768	X	CONSTANTS AND WORKING AREA
11388	26 11381 11387	TF	SET 68 IN P FIELD
11400	24 11381 11385	C	COMPARE 87 TO 68
11412	46 11472 01100	BI	CHECK H/P TRIG FOR NOT H/P
11424	46 11472 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
11436	49 11520	B	
11448	41	NOP	
11460	41	NOP	
ERROR ROUTINE			
11472	46 11496 00100	BI	
11484	39 11509 00100	WA	
11496	47 11520 00300	BNI	
11508	48 70767 5 07	H	
11520	46 11388 00200	BI	
11532	49 11556	B	

ROUTINE 066
CHECK COMPARE FOR NOT H/P NOT E/Z

11544	7958	X	CONSTANTS AND WORKING AREA
11556	26 11549 11553	TF	SET -79 IN P FIELD
11568	24 11549 11555	C	COMPARE -58 TO -79
11580	46 11640 01100	BI	CHECK H/P TRIG FOR NOT H/P
11592	46 11640 01200	BI	CHECK E/Z TRIG FOR NOT E/Z
11604	49 11688	B	
11616	41	NOP	
11628	41	NOP	
			ERROR ROUTINE
11640	46 11664 00100	BI	
11652	39 11677 00100	WA	
11664	47 11688 00300	BNI	
11676	48 70767 6 07	H	
11688	46 11556 00200	BI	
11700	49 11724	B	

ROUTINE 067
CHECK COMPARE IMMED FOR E/Z

11712	10248	X	CONSTANTS AND WORKING AREA
11724	26 11718 11723	TF	SET 10248 IN P FIELD
11736	14 11718 10248	CM	COMPARE IMMED
11748	47 11796 01200	BNI	CHECK E/Z TRIG FOR E/Z
11760	49 11844	B	
11772	41	NOP	
11784	41	NOP	
			ERROR ROUTINE
11796	46 11820 00100	BI	
11808	39 11833 00100	WA	
11820	47 11844 00300	BNI	
11832	48 70767 7 07	H	
11844	46 11724 00200	BI	
11856	49 11916	B	

ROUTINE 068
CHECK ADD TEN DIGIT NO TO 12 DIGIT NO

11868	00	12345	67890	X	AUGEND
11880		23456	78901	X	ADDEND
11892	00	35802	46791	X	COMPARE DATA
11904				X	WORKING AREA
11916	26	11915	11879	TF	SET AUGEND
11928	21	11915	11891	A	ADD ADDEND TO AUGEND
11940	24	11915	11903	C	CHECK FOR CORRECT ANSWER
11952	47	11976	01200	BNI	CHECK E/Z TRIG FOR E/Z
11964	49	12024		B	ERROR ROUTINE
11976	46	12000	00100	BI	
11988	39	12013	00100	WA	
12000	47	12024	00300	BNI	
12012	48	70767	8 0#	H	
12024	46	11916	00200	BI	
12036	49	12096		B	

ROUTINE 069
CHECK SUBT TEN DIGIT NO FROM 12 DIGIT NO

12048	00	98765	43210	X	MINUEND
12060		12345	67890	X	SUBTRAHEND
12072	00	86419	75320	X	COMPARE DATA
12084				X	WORKING AREA
12096	26	12095	12059	TF	SET MINUEND
12108	22	12095	12071	S	SUBT SUBTRAHEND FROM MINUEND
12120	24	12095	12083	C	CHECK FOR CORRECT ANSWER
12132	47	12156	01200	BNI	CHECK E/Z TRIG FOR E/Z
12144	49	12204		B	ERROR ROUTINE
12156	46	12180	00100	BI	
12168	39	12193	00100	WA	
12180	47	12204	00300	BNI	
12192	48	70767	9 0#	H	
12204	46	12096	00200	BI	
12216	49	12240		B	

ROUTINE 070
CHECK MULTIPLY

12228	0121	1111	X	MULTIPLICAND, MULTIPLIER, COMP. DATA	
12240	23	12237	12239	M	MULTIPLY
12252	24	12234	00099	C	CHECK PRODUCT CORRECT
12264	47	12288	01200	BNI	CHECK E/Z TRIG FOR E/Z
12276	49	12336		B	ERROR ROUTINE
12288	46	12312	00100	BI	
12300	39	12325	00100	WA	
12312	47	12336	00300	BNI	
12324	48	70777	0 0#	H	
12336	46	12240	00200	BI	
12348	49	12372		B	

ROUTINE 071
CHECK MULTIPLY

12360	0484	2222	X	MULTIPLICAND, MULTIPLIER, COMP, DATA	
12372	23	12369	12371	M	MULTIPLY
12384	24	12366	00099	C	CHECK PRODUCT CORRECT
12396	47	12420	01200	BNI	CHECK E/Z TRIG FOR E/Z
12408	49	12468		B	ERROR ROUTINE
12420	46	12444	00100	BI	
12432	39	12457	00100	WA	
12444	47	12468	00300	BNI	
12456	48	70777	1 0#	H	
12468	46	12372	00200	BI	
12480	49	12504		B	

ROUTINE 072
CHECK MULTIPLY

12492	1936	4444	X	MULTIPLICAND, MULTIPLIER, COMP. DATA
12504	23	12501 12503	M	MULTIPLY
12516	24	12498 00099	C	CHECK PRODUCT CORRECT
12528	47	12552 01200	BNI	CHECK E/Z TRIG FOR E/Z
12540	49	12600	B	ERROR ROUTINE
12552	46	12576 00100	BI	
12564	39	12589 00100	WA	
12576	47	12600 00300	BNI	
12588	48	70777 2 0#	H	
12600	46	12504 00200	BI	
12612	49	12636	B	

ROUTINE 073
CHECK MULTIPLY

12624	7744	8888	X	MULTIPLICAND, MULTIPLIER, COMP. DATA
12636	23	12633 12635	M	MULTIPLY
12648	24	12630 00099	C	CHECK PRODUCT CORRECT
12660	47	12684 01200	BNI	CHECK E/Z TRIG FOR E/Z
12672	49	12732	B	ERROR ROUTINE
12684	46	12708 00100	BI	
12696	39	12721 00100	WA	
12708	47	12732 00300	BNI	
12720	48	70777 3 0#	H	
12732	46	12636 00200	BI	
12744	49	12768	B	

ROUTINE 074
CHECK MULTIPLY IMMED

12756	00000 0 777	X	MULTIPLICAND
12768	13 12767 00000	MM	MULTIPLY IMMED
12780	47 12840 01200	BNI	CHECK E/Z TRIG FOR E/Z
12792	24 12763 00099	C	CHECK PRODUCT CORRECT
12804	47 12840 01200	BNI	CHECK E/Z TRIG FOR E/Z
12816	49 12888	B	
12828			

ERROR ROUTINE

12840	46 12864 00100	BI	
12852	39 12877 00100	WA	
12864	47 12888 00300	BNI	
12876	48 70777 4 0#	H	
12888	46 12768 00200	BI	
12900	49 12960	B	

ROUTINE 075
CHECK MULTIPLY

12912	01234 56789	X	MULTIPLICAND
12924	01234 56789	X	MULTIPLIER
12936	000 15241	X	COMPARE DATA
12948	57 87501 90521	X	COMPARE DATA
12960	23 12923 12935	M	MULTIPLY
12972	24 12959 00099	C	CHECK PRODUCT CORRECT
12984	47 13008 01200	BNI	CHECK E/Z TRIG FOR E/Z
12996	49 13056	B	

ERROR ROUTINE

13008	46 13032 00100	BI	
13020	39 13045 00100	WA	
13032	47 13056 00300	BNI	
13044	48 70777 5 0#	H	
13056	46 12960 00200	BI	
13068	49 13128	B	

ROUTINE 076
CHECK MULTIPLY

13080	37	92022	34363	X	MULTIPLICAND
13092	82	06972	21257	X	MULTIPLIER
13104	31	12102	20036	X	COMPARE DATA
13116	15	97794	54291	X	COMPARE DATA
13128	16	00079	00000	TFM	SET MEM POS 75-79 TO ZERO
13140	23	13091	13103	M	MULTIPLY
13152	24	00099	13127	C	CHECK PRODUCT
13164	47	13200	01200	BNI	CHECK E/Z TRIG FOR E/Z
13176	44	13200	00076	BNF	CHECK HIGH ORDER POS FOR FLAG
13188	49	13248		B	ERROR ROUTINE
13200	46	13224	00100	BI	
13212	39	13237	00100	WA	
13224	47	13248	00300	BNI	
13236	48	70777	6 07	H	
13248	46	13128	00200	BI	
13260	49	13992		B	

ROUTINE 077
CHECK CONTROL OPERATIONS &
WRITE NUM & ALPHA

13272	12	345#6	78907	X	NUMERIC DATA
13284	55	6454	4955	X	ALPHA DATA NUM IN
13296	46	56	4 14256	X	ALPHA DATA FO ABO
13308	65	45	5 64646	X	ALPHA DATA VE OFF
13320	62	4563	6356	X	ALPHA DATA SET TO
13332		59494	74863	X	ALPHA DATA RIGHT
13344		63665	6 62	X	ALPHA DATA TWO S
13356	57	41434	562	X	ALPHA DATA PACES
13368	42	45636	64545	X	ALPHA DATA BETWEEN
13380	55	75	4155	X	ALPHA DATA N 5 AN
13392	44	76	6348	X	ALPHA DATA D 6 TH
13404	59	4545	5349	X	ALPHA DATA REE LI
13416	55	4562	5646	X	ALPHA DATA NES OF
13428		44416	34107	X	ALPHA DATA DATA.
13440	46	13764	00100	BI	CHECK SW 1 FOR TYPEOUT

13452	34	00102	K	CARRIAGE RETURN	
13464	34	00108	K	TAB	
13476	38	13272	00100	WN	TYPEWRITER
13488	34	00101	K	SPACE	
13500	34	00101	K	SPACE	
13512	38	13278	00100	WN	TYPEWRITER
13524	34	00102	K	CARRIAGE RETURN	
13536	34	00108	K	TAB	
13548	38	13272	00100	WN	TYPEWRITER
13560	34	00101	K	SPACE	
13572	34	00101	K	SPACE	
13584	38	13278	00100	WN	TYPEWRITER
13596	34	00101	K	SPACE	
13608	34	00101	K	SPACE	
13620	38	13272	00100	WN	TYPEWRITER
13632	34	00102	K	CARRIAGE RETURN	
13644	34	00108	K	TAB	
13656	38	13272	00100	WN	TYPEWRITER
13668	34	00101	K	SPACE	
13680	34	00101	K	SPACE	
13692	38	13278	00100	WN	TYPEWRITER
13704	34	00102	K	CARRIAGE RETURN	
13716	39	13285	00100	WA	TYPEWRITER
13728	34	00102	K	CARRIAGE RETURN	
13740	46	13440	00200	BI	CHECK SW 2 FOR LOOP ROUTINE
13752	49	13764	B		

ROUTINE 078
CHECK DUMP NUMERIC TO
TYPEWRITER & PAPER TAPE PUNCH
FOR CARD I/O, SEE PAGE 55A

13764	34	00102	K	CARRIAGE RETURN	
13776	35	19976	00100	DN	DUMP NUMERIC-TYPEWRITER
13788	35	19976	00200	DN	DUMP NUMERIC-TAPE PUNCH
13800	49	13944	B		
13812	41		NOP		

ROUTINE 078
Check Dump Numeric to Typewriter
& Card Punch. For Paper Tape,
See Page 55

13764	34		00102	K	Carriage Return
13776	35	19976	00100	DN	Dump Numeric to Typewriter
13788	35	19920	06400	DN	Dump Numeric to Card Punch
13800	49	13932		B	
13812	41			NOP	

ROUTINE 079
Check WA, Punched Data will then
be read in
For Paper Tape, See Page 56

13824	03	04101	31420	X	.)+\$*-
13836	21	23243	33400	X	/, (=@
13848	41	42434	44546	X	ABCDEF
13860	47	48495	15253	X	GHJKLM
13872	54	55565	75859	X	MNOPQR
13884	62	63646	56667	X	STUVWX
13896	68	69707	17273	X	YZ0123
13908	74	75767	77879	X	456789
13920	07			X	£
13932	31	16044	13824	TR	Transmit Data to Punch Area
13944	39	16045	00400	WA	Card Punch
13956	39	16045	00400	WA	Card Punch
13968	39	16045	00400	WA	Card Punch
13980	49	19924		B	

Check if Division Installed

13992	49	18808		B	
14004	16	13998	14052	TFM	
14016	49	00552		B	

ROUTINE 079
CHECK WA. PUNCHED DATA WILL THEN BE READ IN

13824	03	04101	31420	X	.)+\$*-
12836	21	23243	33400	X	/, (=@
13848	41	42434	44546	X	ABCDEF
13860	47	48495	15253	X	GHIJKL
13872	54	55565	75859	X	MNOPQR
13884	62	63646	56667	X	STUVWX
13896	68	69707	17273	X	YZ0123
13908	74	75767	77879	X	456789
13920	0#			X	#
13932				X	
13944	39	13825	00200	WA	PAPER TAPE PUNCH
13956	39	13825	00200	WA	PAPER TAPE PUNCH
13968	39	13825	00200	WA	PAPER TAPE PUNCH
13980	49	19924		B	
					CHECK IF DIVISION INSTALLED
13992	49	18808		B	
14004	16	13998	14052	TFM	
14016	49	00552		B	

ROUTINE 080
CHECK LOAD DIVIDEND

14028	45	67890	12304	X	DIVIDEND
14040	00	00000	00000	X	COMPARE DATA
14052	28	00095	14039	LD	LOAD DIVIDEND
14064	24	14043	00099	C	COMP DIVIDEND WITH COMP DATA
14076	47	14136	01200	BNI	CHECK E/Z TRIG FOR E/Z
14088	32	00080	00000	SF	SET FLAG POS. 80
14100	24	14047	00083	C	CHECK FOR ZERO POS. 80-83
14112	47	14136	01200	BNI	CHECK E/Z TRG FOR E/Z
14124	49	14184	00000	B	
					ERROR ROUTINE
14136	46	14160	00100	BI	
14148	39	14173	00100	WA	
14160	47	14184	00300	BNI	
14172	48	70787	0 0#	H	
14184	46	14052	00200	BI	
14196	49	14244		B	

ROUTINE 081
CHECK LOAD DIVIDEND

14208	I2	34567	89086	X	DIVIDEND
14220	I2	34567	89086	X	COMPARE DATA
14232	00	00000	0	X	COMPARE DATA
14244	28	00091	14219	LD	LOAD DIVIDEND
14256	24	14239	00099	C	COMP DIVIDEND WITH COMP DATA
14268	47	14304	01200	BNI	CHECK E/Z TRIG FOR E/Z
14280	49	14352		B	
14292	41			NOP	
ERROR ROUTINE					
14304	46	14328	00100	BI	
14316	39	14341	00100	WA	
14328	47	14352	00300	BNI	
14340	48	70787	1 07	H	
14352	46	14244	00200	BI	
14364	49	14488		B	
14376				X	
14388				X	
14400				X	
14412				X	
14424				X	
14436				X	
14448				X	
14460				X	
14472				X	

ROUTINE 082
CHECK LOAD DIVIDEND IMMED

14476	78693	00000		X	COMPARE DATA
14488	18	00094	78693	LDM	LOAD DIVIDEND IMMED
14500	24	14487	00099	C	COMP DIVIDEND WITH COMP DATA
14512	47	14488	01200	BNI	CHECK E/Z TRIG FOR E/Z
14524	49	14596		B	
14536	41			NOP	
ERROR ROUTINE					
14548	46	14572	00100	BI	
14560	39	14585	00100	WA	
14572	47	14596	00300	BNI	
14584	48	70787	2 07	H	
14596	46	14488	00200	BI	
14608	49	14644		B	

ROUTINE 083
CHECK DIVIDE

14620	12	34567	89123	X	DIVIDEND, DIVISOR
14632	45	10000	06789	X	DIVISOR, COMPARE DATA
14644	28	00099	14628	LD	LOAD DIVIDEND
14656	29	00094	14633	D	DIVIDE
14668	24	14638	00094	C	COMP. QUOTIENT TO COMP DATA
14680	47	14740	01200	BNI	CHECK E/Z TRIG FOR E/Z
14692	24	14643	00099	C	COMP REMAINDER TO COMP DATA
14704	47	14740	01200	BNI	CHECK E/Z TRIG FOR E/Z
14716	49	14788		B	
14728	41			NOP	ERROR ROUTINE
14740	46	14764	00100	BI	
14752	39	14777	00100	WA	
14764	47	14788	00300	BNI	
14776	48	70787	3 07	H	
14788	46	14644	00200	BI	
14800	49	14836		B	

ROUTINE 084
CHECK DIVIDE

14812	98	76543	21678	X	DIVIDEND, DIVISOR
14824	91	45478	4179	X.	DIVISOR, QUOTIENT, REMAINDER
14836	28	00099	14820	LD	LOAD DIVIDEND
14848	29	00094	14824	D	DIVIDE
14860	24	14830	00095	C	COMPARE QUOTIENT
14872	47	14932	01200	BNI	CHECK E/Z TRIG FOR E/Z
14884	24	14834	00099	C	COMPARE REMAINDER
14896	47	14932	01200	BNI	CHECK E/Z TRIG FOR E/Z
14908	49	14980		B	
14920	41			NOP	
ERROR ROUTINE					
14932	46	14956	00100	BI	
14944	39	14969	00100	WA	
14956	47	14980	00030	BNI	
14968	48	70787	4 07	H	
14980	46	14836	00200	BI	
14992	49	15028		B	

ROUTINE 085
CHECK DIVIDE

15004	98	76543	21123	X	DIVIDEND, DIVISOR
15016	45	80004	04941	X	DIVISOR, QUOTIENT, REMAINDER
15028	28	00099	15012	LD	LOAD DIVIDEND
15040	29	00095	15017	D	DIVIDE
15052	24	15022	00094	C	COMPARE QUOTIENT
15064	47	15124	01200	BNI	CHECK E/Z TRIG FOR E/Z
15076	24	15027	00099	C	COMPARE REMAINDER
15088	47	15124	01200	BNI	CHECK E/Z TRIG FOR E/Z
15100	49	15172		B	
15112	41			NOP	

ERROR ROUTINE

15124	46	15148	00100	BI
15136	39	15161	00100	WA
15148	47	15172	00300	BNI
15160	48	70787	5 07	H
15172	46	15028	00200	BI
15184	49	15220		B

ROUTINE 086

CHECK DIVIDE

15196	<u>67</u>	84219	53476	X	DIVIDEND, DIVISOR
15208	21	42465	3623	X	DIVISOR, QUOTIENT, REMAINDER
15220	28	00099	15204	LD	LOAD DIVIDEND
15232	29	00084	15208	D	DIVIDE
15244	24	00095	15214	C	COMPARE QUOTIENT
15256	47	15316	01200	BNI	CHECK E/Z TRIG FOR E/Z
15268	24	15218	00099	C	COMPARE REMAINDER
15280	47	15316	01200	BNI	CHECK E/Z TRIG FOR E/Z
15292	49	15364		B	
15304	41			NOP	

ERROR ROUTINE

15316	46	15340	00100	BI
15328	39	15353	00100	WA
15340	47	15364	00300	BNI
15352	48	70787	6 07	H
15364	46	15220	00200	BI
15376	49	15400		B

ROUTINE 087
CHECK DIVIDE IMMEDIATE

15388	09020 0	X	QUOTIENT, REMAINDER
15400	18 00099 86592	LDM	LOAD DIVIDEND IMMEDIATE
15412	19 00096 00096	DM	DIVIDE IMMEDIATE
15424	24 15393 00097	C	COMPARE QUOTIENT
15436	47 15496 01200	BNI	CHECK E/Z TRIG FOR E/Z
15448	24 15395 00099	C	COMPARE REMAINDER
15460	47 15496 01200	BNI	CHECK E/Z TRIG FOR E/Z
15472	49 15544	B.	
15484	41	NOP	ERROR ROUTINE
15496	46 15520 00100	BI	
15508	39 15533 00100	WA	
15520	47 15544 00300	BNI	
15532	48 70787 7 07	H	
15544	46 15400 00200	BI	
15556	49 15568	B	

ROUTINE 088
CHECK DIVIDE BY ZERO INDICATION

15568	18 00096 39486	LDM	LOAD DIVIDEND IMMEDIATE
15580	19 00096 00000	DM	DIVIDE IMMEDIATE
15592	47 15628 01400	BNI	CHECK FOR OVERFLOW ON
15604	49 15676	B	
15616	41	NOP	ERROR ROUTINE
15628	46 15652 00100	BI	
15640	39 15665 00100	WA	
15652	47 15676 00300	BNI	
15664	48 70787 8 07	H	
15676	46 15568 00200	BI	
15688	49 15700	B	

ROUTINE 089
CHECK OVERFLOW INDIC. FIRST DIGIT GREATER ZERO

15700	18	00096	34278	LDM	LOAD DIVIDEND IMMEDIATE
15712	19	00095	00314	DM	DIVIDE IMMEDIATE
15724	47	15760	01400	BNI	CHECK FOR OVERFLOW ON
15736	49	15808		B	
15748	41			NOP	
ERROR ROUTINE					
15760	46	15784	00100	BI	
15772	39	15797	00100	WA	
15784	47	15808	00300	BNI	
15796	48	70787	9 0#	H	
15808	46	15700	00200	BI	
15820	49	18808		B	
15832	41			NOP	

ROUTINE 090
TIMES 1000 ROUTINE AND CHK NOP

15844			000	X	CONSTANTS AND WORKING AREA
15856	41			NOP	
15868	46	15916	01400	BI	TURN OFF OVERFLOW
15880	11	15855	00001	AM	ADD ONE TO P FIELD
15892	46	13440	01400	BI	CHECK FOR OVERFLOW
15904	49	01116		B	
ERROR ROUTINE					
15916	46	15940	00100	BI	
15928	39	15953	00100	WA	
15940	47	15856	00300	BNI	
15952	48	70797	0 #	H	
15964	49	15856		B	

ROUTINE 097
BRANCH INDICATOR CHECK

18796	00	00000	T1122		Working Area
18808	26	18855	18802	TF	Clear Math Area
18820	21	18855	18802	A	Add 00 to 00 Causing E/Z, H/P or E/Z, Not H/P, and No O/F
18832	46	18856	01300	BI	Check BI H/P or E/Z for H/P or E/Z
18844	49	19228		B	Branch to Error Routine
18856	47	19228	01300	BNI	Check BNI H/P or E/Z for H/P or E/Z
18868	47	18892	01100	BNI	Check BNI H/P for Not H/P
18880	49	19228		B	Branch to Error Routine
18892	46	19228	01100	BI	Check BI H/P for Not H/P
18904	46	18928	01200	BI	Check BI E/Z for E/Z
18916	49	19228		B	Branch to Error Routine
18928	47	19228	01200	BNI	Check BNI E/Z for E/Z
18940	47	18964	01400	BNI	Check BNI O/F for No O/F
18952	49	19228		B	Branch to Error Routine
18964	46	19228	01400	BI	Check BI O/F for No O/F
18976	14	18984	00#10	CM	Compare 10 to -00 Causing Not H/P, Not E/Z, Not H/P or E/Z, and No O/F
18988	47	19012	01200	BNI	Check BNI E/Z for Not E/Z
19000	49	19228		B	Branch to Error Routine
19012	46	19228	01200	BI	Check BI E/Z for Not E/Z
19024	47	19048	01300	BNI	Check BNI H/P or E/Z for Not H/P or E/Z
19036	49	19228		B	Branch to Error Routine
19048	46	19228	01300	BI	Check BI H/P or E/Z for Not H/P or E/Z
19060	21	18855	18804	A	Add 11 to 00 Causing H/P, H/P or E/Z, Not E/Z, and No
19072	46	19096	01100	BI	Check BI H/P for H/P
19084	49	19228		B	Branch to Error Routine
19096	47	19228	01100	BNI	Check BNI H/P for H/P
19108	46	19132	01300	BI	Check BI H/P or E/Z for H/P or E/Z
19120	49	19228		B	Branch to Error Routine
19132	47	19228	01300	BNI	Check BNI H/P or E/Z for H/P or E/Z
19144	21	18855	18805	A	Add 111 to 11 Causing O/F, H/P, H/P or E/Z, and Not E/Z
19156	46	19180	01400	BI	Check BI O/F for O/F
19168	49	19228		B	Branch to Error Routine
19180	22	18855	18805	A	Add 111 to 22 Causing O/F, H/P, H/P or E/Z, and Not E/Z
19192	47	19228	01400	BNI	Check BNI O/F for O/F
19204	47	19276	01400	BNI	
19216	49	19228		B	
ERROR ROUTINE					
19228	46	19252	00100	BI	
19240	39	19265	00100	WA	
19252	47	19276	00300	BNI	
19264	48	70797	7000#	H	
19276	46	18808	00200	BI	
19288	49	15856		B	
19300					

ROUTINE 099
CHECK TAPE OUTPUT. READ IN TAPE THEN TYPE
FOR CARD I/O, SEE PAGE 63A

16044		X
16056		X
16068		X
16080		X
16092		X
16104		X
16116		X
16128		X
16140		X
16152		X
16164		X
16176		X
16188		X
16200		X
16212		X
16224		X
16236	48	H
16248	41	NOP
16260	34	00102
16272	36	16124 00300
16284	38	16124 00100
16296	38	16140 00100
16308	34	00102
16320	37	16069 00300
16332	39	16069 00100
16344	34	00102
16356	37	16069 00300
16368	39	16069 00100
16380	34	00102
16392	37	16069 00300
16404	39	16069 00100
16416	34	00102
16428	44	16488 16152
16440	44	16488 16153
16452	49	16500
16464		X
16476		X
16488	39	16501 00100
16500	48	70797 9000#
16512	49	00552

ERROR ROUTINE

16488	39	16501 00100	WA
16500	48	70797 9000#	H
16512	49	00552	B

ROUTINE 098
TEST COMPLETED ROUTINE

19720	63	45626 3 59	X	TEST R
19732	56	64634 95545	X	OUTLINE
19744	62	435 65457	X	S COMP
19756	53	45634 54403	X	LETED.
19768		4946 6266	X	IF SW
19780	71	564 646	X	I OFF
19792	41	5544 5556	X	AND NO
19804		59566 46349	X	ROUTI
19816	55	45 5 55662	X	NE NOS
19828		63685 74544	X	TYPED
19840		56646 323	X	OUT

ROUTINE 099
CHECK CARD OUTPUT. READ IN THEN TYPE
(FOR PAPER TAPE I/O, SEE PAGE 63)

16044		X
16056		X
16068		X
16080		X
16092		X
16104		X
16116		X
16128		X
16140		X
16152		X
16164		X
16176		X
16188		X
16200		X
16212		X
16224		X
16236	48	H
16248	15 16148	# TDM
16260	34	00102 K
16272	36	16068 00500 RN
16284	38	16124 00100 WN
16296	38	16140 00100 WN
16308	34	00102 K
16320	37	16069 00500 RA
16332	39	16069 00100 WA
16344	34	00102 K
16356	37	16069 00500 RA
16368	39	16069 00100 WA
16380	34	00102 K
16392	37	16069 00500 RA
16404	39	16069 00100 WA
16416	34	00102 K
16428	44	16488 16152 BNF
16440	44	16488 16153 BNF
16452	49	16500 B
16464		X
16476		X
ERROR ROUTINE		
16488	39	16501 00100 WA
16500	48	70797 9000# H
16512	49	00552 B

19852	54	41434	84955	X	MACHIN
19864	45	574	55946	X	E PERF
19876	56	59544	544	X	ORMED
19888	63	45626	362	X	TESTS
19900	57	59565	74559	X	PROPER
19912	53	68030	7	X	LY.
19924	34	00102		K	
19936	39	19721	00100	WA	
19948	46	00552	00400	BI	
19960	49	16236		B	
19972	199	76012		X	DUMP NUMERIC DATA
19984	34	56789	71219	X	DUMP NUMERIC DATA
19996	99	89	L	X	DUMP NUMERIC DATA

CU05– Special Instructions Diagnostic


```
//=====
//  
// CU05 - Special Instructions Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Bypass error type out  
// OFF - Type out routine number on error  
// PS2: ON - Loop in routine  
// OFF - Continue to next routine  
// PS3: ON - Stop on error  
// OFF - Do not stop on error, continue  
// PS4: ON - Repeat test CU05  
// OFF - Run test CU05 once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - PROGRAM  
//  
// Start addresses:  
//  
// 00828 - Full test  
//  
// Directions:  
//  
// 1. Load CU05 diagnostic  
// 2. Press START  
// 3. Press START  
//=====
```


Sample Output - CU05

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SENSE SWS FOR CU05, SPECIAL INSTS TEST.
THEN START.

START ROUTINES. ETOS FOLLOW.

TEST COMPLETED. IF SW 1 OFF AND NO ROUTINE NOS TYPED, TEST PERFORMED PROPERLY.

NO. 2125637
SHEET 0
OF 18

DIAGNOSTIC TEST

TITLE 1620 SPECIAL INSTRUCTIONS DIAGNOSTIC TEST - CU05
MACH. TYPE 1620 BY J.H.M. APPR. G.I.A. DATE 4-11-62

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404618	5-15-61	1-18
404675	4-11-62	2A, 3A, 6, 6A, 7, 8, 9, 10, 12, 14, 15, 16, 17, 17A, 18

E/C NO.	404618	404675					
DATE	5-15-61	4-11-62					

1620 DIAGNOSTICS

CU05

SPECIAL INSTRUCTIONS

A. SCOPE:

This test is essentially a fault detection test designed to check for the proper operation of the special instructions Transfer Numeric Strip, Transfer Numeric Fill and Move Flag.

B. SET UP:

Suggested setting of 1620 switches:

1. Check Switches set to STOP.
2. Sense Switches set OFF.

The sense switches have the following functions in this test:

Sense Switch #1	ON	Bypass error type out
	OFF	Type out routine number on error
Sense Switch #2	ON	Loop in routine
	OFF	Continue to next routine
Sense Switch #3	ON	Stop on error
	OFF	Bypass Halt in error routine
Sense Switch #4	ON	Repeat Test
	OFF	Halt

NORMAL LOAD

This test is designed to be used with a 1620-1622 system. The cards must be run in the 1622 by using the LOAD key. Depressing the 1620 START key will initiate the loading of the instructions.

If these features are to be used with a 1620-1621 system, then the instructions to load the test instructions must be inserted in the 1620.

These are:

36 00096 00300
49 00828

PRODUCE NEW TAPE

1. Load tape in paper tape reader and ready reader.
2. Insert the following instructions in 1620:

PN 2125637
EC 404618

PRODUCE NEW TAPE (cont'd)

36 14960 00300
35 14960 00200
37 15201 00300
39 15201 00200
48

3. Release and start.

C. TEST METHOD:

The test is comprised of a group of routines (15) to check the various aspects of these three special instructions.

Routine 001 types the setting of the sense switches, the name of the test, and reads in the data for checking Transfer Numeric Strip.

Routines 002-005 check the Transfer Numeric Strip circuits. The zone data of the numeric fields read in in the alpha mode in routine 001 is removed. Routines 002 and 003 remove the zone data from positive fields. The addressed position of the Q field in routine 002 is odd; while the addressed position of the Q field in routine 003 is even. Routines 004 and 005 remove the zone data from negative fields; the addressed position of the Q fields being odd and even respectively.

Routines 006-009 check the Transfer Numeric Fill Circuits. Zone data for ten digit numeric fields is added. Routines 006 and 007 add zone data to positive fields. The addressed Q field for routine 006 is an odd memory location and for routine 007 it is an even position. A test is made to determine that a flag bit is not inserted in the high order position of the P field. Routines 008 and 009 add zone data to negative fields. The addressed Q field in routine 008 is odd and in routine 009 it is even. Flag bits should not be inserted in the high order position of the P field.

Routines 010-013 check the Move Flag circuits. The flag conditions of the Q addressed position is moved to the flag position of the P address, leaving the Q address without a flag. The four conditions, moving a flag to a no flag position, flag to flag position, no flag to flag, and no flag to no flag, are checked by the routines 010-013 respectively.

Routine 014 is the repeat routine, and routine 015 is the test completed routine.

Error typeouts consist of an "H" followed by the routine number and a statement of the error that occurred or the field that is in question. Comparing the typed field with the compare data will indicate what caused the error type out.

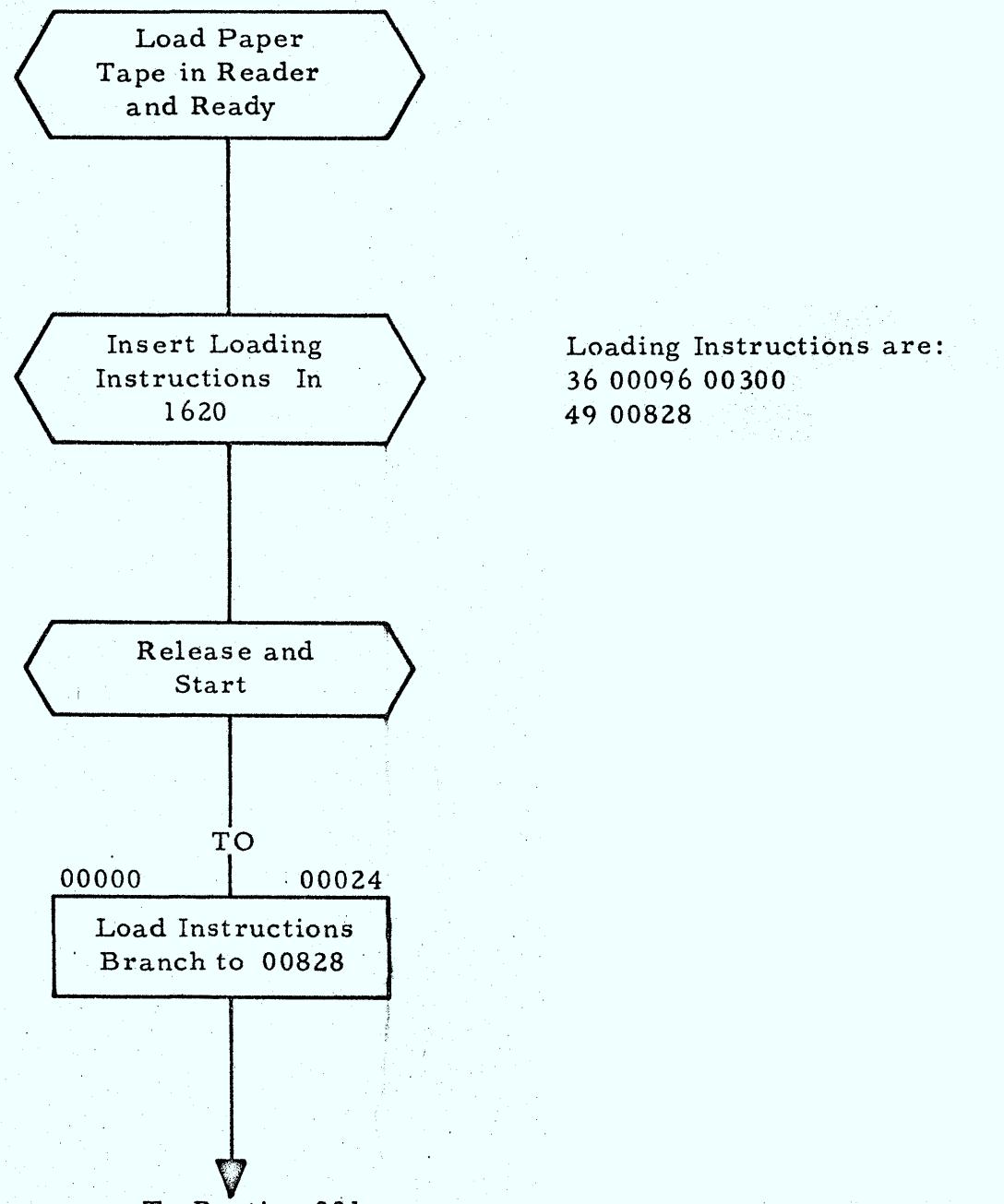
PN 2125637 EC 404618

REZ 19396

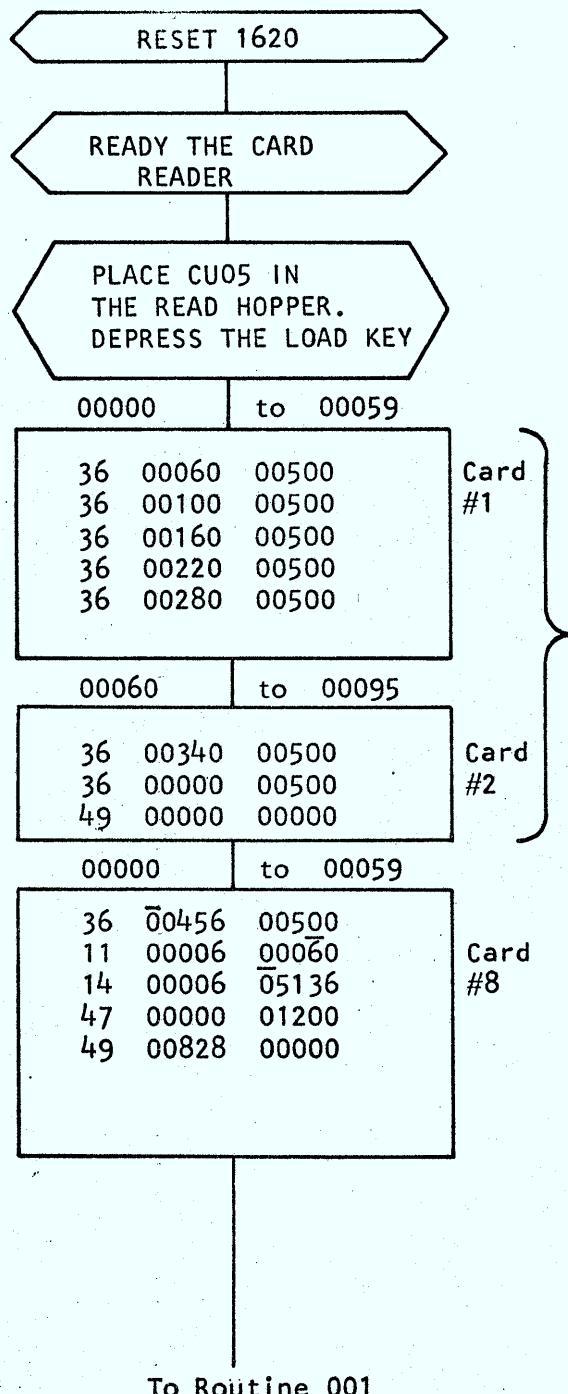
The complete normal typeout information will be as follows:

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SENSE SWS FOR
CU05 SPECIAL INSTS TEST. THEN START.
START ROUTINES. ETOS FOLLOW,
TEST COMPLETED. IF SW 1 OFF AND NO ROUTINE NOS TYPED,
TEST PERFORMED PROPERLY.

1620 DIAGNOSTICS
CU0\$
SPECIAL INSTRUCTIONS
FLOW CHART

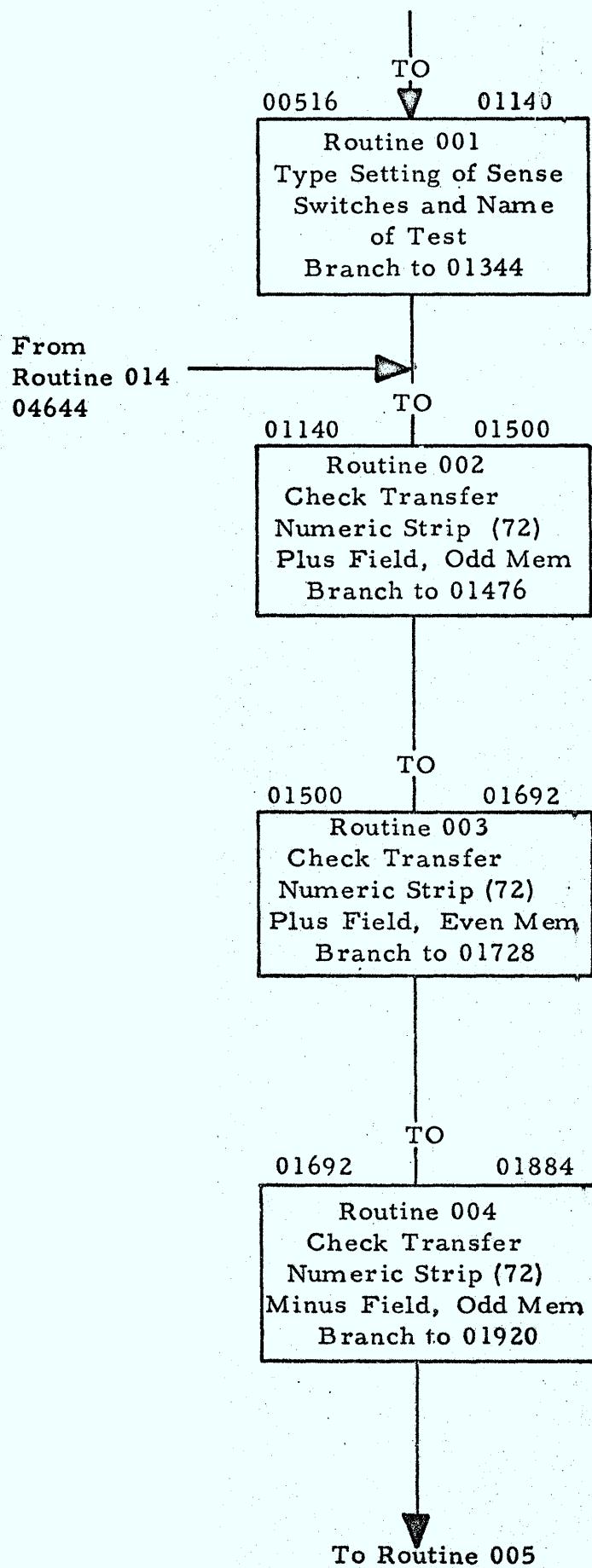


PN 2125637
EC 404618

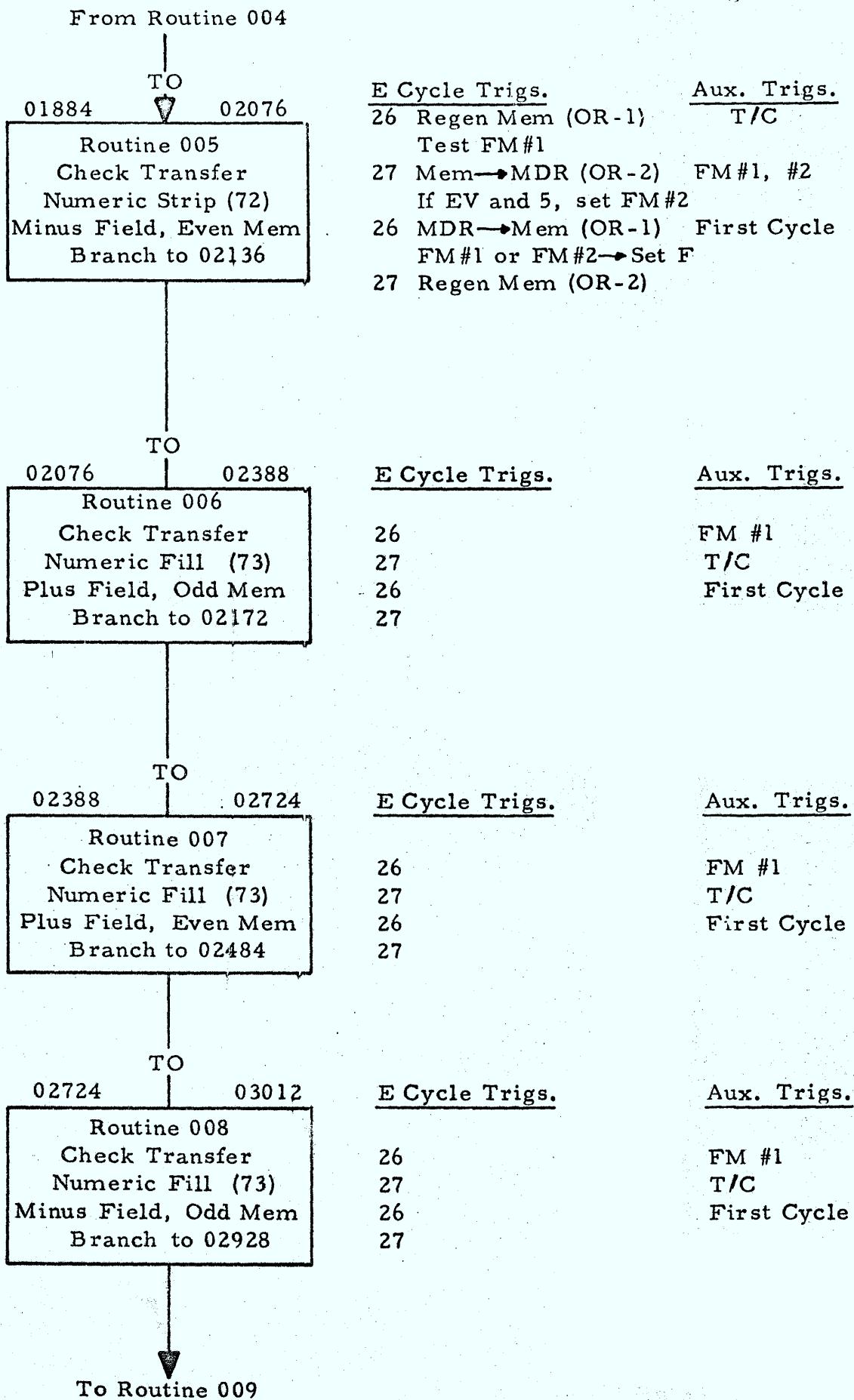
CU05 FLOW CHART
WITH 1622 I/O

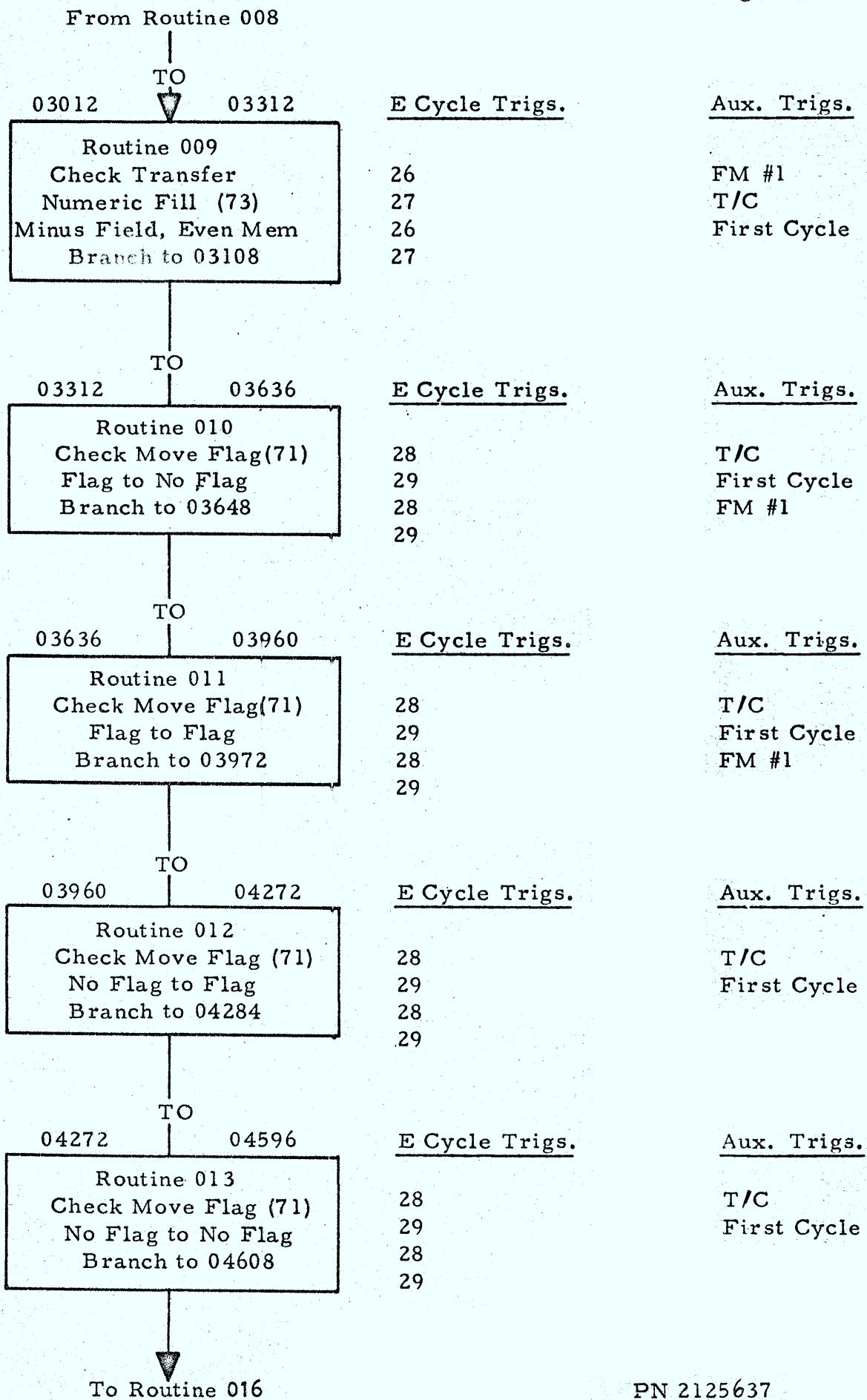
First and second Load Cards load the
math tables and Program Load Card.
(Cards 3 through 7 contain the math tables.)

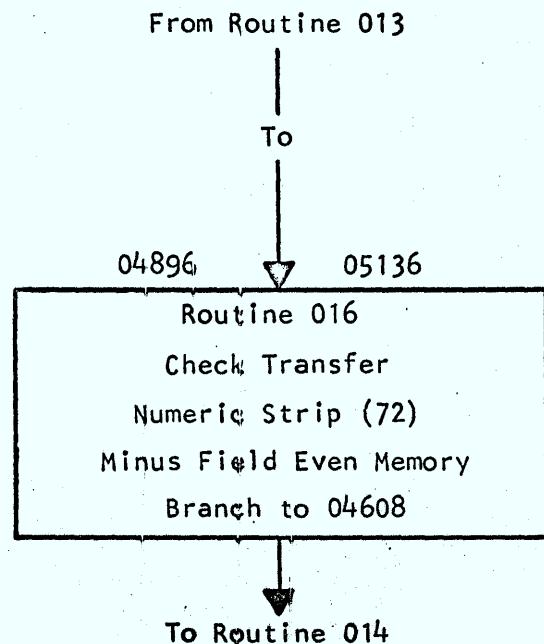
Eighth Load Card contains instructions
for loading core storage.

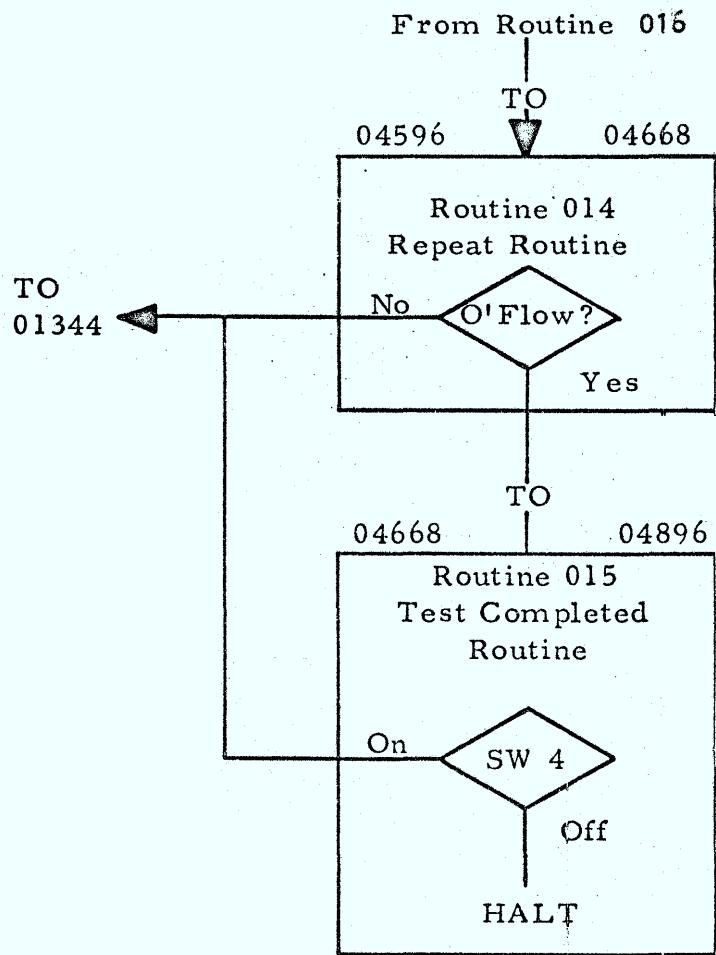


<u>E Cycle Trigs.</u>	<u>Aux. Trigs.</u>
26 Regen Mem (OR-1)	T/C
Test for FM#1	
27 Mem → MDR (OR-1)	FM #1
26 MDR → Mem (OR-1)	First Cycle
FM#1 → Set F	
27 Regen Mem	
<u>E Cycle Trigs.</u>	<u>Aux. Trigs.</u>
26 Regen Mem (OR-1)	T/C
Test FM#1	
27 Mem → MDR (OR-2)	FM #1
26 MDR → Mem (OR-1)	First Cycle
FM#1 → Set F	
27 Regen Mem (OR-2)	
<u>E Cycle Trigs.</u>	<u>Aux. Trigs.</u>
26 Regen Mem (OR-1)	T/C
Test FM#1	
27 Mem → MDR (OR-2)	FM#1, #2
If EV and 5, set FM#2	
26 MDR → Mem (OR-1)	First Cycle
FM#1 or FM#2 → Set F	
27 Regen Mem (OR-2)	









PN 2125637
EC 404675

1620 DIAGNOSTICS
CU05
SPECIAL INSTRUCTIONS

Mem Loc	00 01	PPPPP 23456	QQQQQ 78901	OP TYP	DESCRIPTION
00096		0000	00000	MT	MULTIPLY TABLE
00108	00	00102	03040	MT	MULTIPLY TABLE
00120	00	20406	08000	MT	MULTIPLY TABLE
00132	30	60902	10040	MT	MULTIPLY TABLE
00144	80	21610	05001	MT	MULTIPLY TABLE
00156	51	02006	02181	MT	MULTIPLY TABLE
00168	42	00704	11282	MT	MULTIPLY TABLE
00180	00	80614	22300	MT	MULTIPLY TABLE
00192	90	81726	30000	MT	MULTIPLY TABLE
00204	00	00005	06070	MT	MULTIPLY TABLE
00216	80	90012	14161	MT	MULTIPLY TABLE
00228	81	51811	24272	MT	MULTIPLY TABLE
00240	02	42822	36352	MT	MULTIPLY TABLE
00252	03	53045	40363	MT	MULTIPLY TABLE
00264	24	84455	32494	MT	MULTIPLY TABLE
00276	65	36048	46546	MT	MULTIPLY TABLE
00288	27	54453	62718	MT	MULTIPLY TABLE
00300	01	23456	78912	AT	ADD TABLE
00312	34	56789	02345	AT	ADD TABLE
00324	67	89013	45678	AT	ADD TABLE
00336	90	12456	78901	AT	ADD TABLE
00348	23	56789	01234	AT	ADD TABLE
00360	67	89012	34578	AT	ADD TABLE
00372	90	12345	68901	AT	ADD TABLE
00384	23	45679	01234	AT	ADD TABLE
00396	56	78#		AT	ADD TABLE
00408				X	
00420				X	
00432				X	
00444				X	
00456				X	
00468				X	
00480			62	X	
00492	63	41596	3 59	X	
00504	56	64634	95545	X	

ROUTINE 001
CHECK SENSE SWITCH SETTINGS

00516	62	03 4	56356	X	ETO
00528	62	465	65353	X	S FOLL
00540	56	6603	0#62	X	OW. #S
00552	66	71	5655	X	W 1 ON
00564		0#626	6 71	X	# SW 1
00576		56464	6 0#	X	OFF #
00588	62	66 7	2 56	X	SW 2 0
00600	55	0#6	266	X	N #SW
00612	72	564	646	X	2 OFF
00624	0#	6266	73	X	# SW 3
00636	56	55 0	#6266	X	ON # SW
00648		73 5	64646	X	3 OFF
00660		0#626	6 74	X	# SW 4
00672		5655	0#62	X	ON #S
00684	66	74	5646	X	W 4 OF
00696	46	0#6	24563	X	F # SET
00708		62455	56245	X	SENSE
00720		62666	2 46	X	SWS F
00732	56	59 4	36470	X	OR CUO
00744	75	23 6	25745	X	5, SPE
00756	43	49415	3 49	X	CIAL I
00768	55	62636	2 63	X	NSTS T
00780	45	62630	3 63	X	EST. T
00792	48	4555	6263	X	HEN ST
00804	41	59630	3 0#	X	ART. #
00816				X	
00828	46	00852	00100	BI	CHECK FOR SW 1 ON
00840	47	00876	00100	BNI	CHECK FOR SW 1 OFF
00852	39	00551	00100	WA	TYPE SW 1 ON
00864	49	00888		B	
00876	39	00569	00100	WA	TYPE SW 1 OFF
00888	46	00912	00200	BI	CHECK FOR SW 2 ON
00900	47	00936	00200	BNI	CHECK FOR SW 2 OFF
00912	39	00589	00100	WA	TYPE SW 2 ON
00924	49	00948		B	
00936	39	00607	00100	WA	TYPE SW 2 OFF
00948	46	00972	00300	BI	CHECK FOR SW 3 ON
00960	47	00996	00300	BNI	CHECK FOR SW 3 OFF
00972	39	00627	00100	WA	TYPE SW 3 ON
00984	49	01008		B	
00996	39	00645	00100	WA	TYPE SW 3 OFF
01008	46	01032	00400	BI	CHECK FOR SW 4 ON
01020	47	01056	00400	BNI	CHECK FOR SW 4 OFF
01032	39	00665	00100	WA	TYPE SW 4 ON
01044	49	01068		B	

01056	39	00683	00100	WA	TYPE SW 4 OFF
01068	39	00703	00100	WA	TYPE SET SWS, NAME OF TEST, THEN START
01080	48			H	HALT
01092	34		00102	K	
01104	39	00491	00100	WA	TYPE START ROUTINES. ETOS FOLLOW
01116	37	01141	00300	RA	READ ALPHA *FOR CARD I/O = 5
01128	49	01344		B	

ROUTINE 002
CHECK TRANSFER NUMERIC STRIP
10 DIGITS + TO ODD MEM.

01140			X	READ CARD AREA	
01152			X	READ CARD AREA	
01164			X	READ CARD AREA	
01176			X	READ CARD AREA	
01188			X	READ CARD AREA	
01200			X	READ CARD AREA	
01212			X	READ CARD AREA	
01224			X	READ CARD AREA	
01236			X	READ CARD AREA	
01248			X	READ CARD AREA	
01260			X	READ CARD AREA	
01272			X	READ CARD AREA	
01284			X	READ CARD AREA	
01296			X	READ CARD AREA	
01308			X	WORKING AREA	
01320	0		X	WORKING AREA	
01332	#	12345	67890	X	COMPARE DATA
01344	72	01159	01331	TNS	TRANSFER NUMERIC STRIP + FIELD
01356	24	01331	01343	C	CHECK FOR CORRECT RESULTS
01368	47	01416	01200	BNI	CHECK FOR E/Z
01380	49	01476		B	
01392			X		
01404			X		
01416	46	01452	00100	BI	ERROR ROUTINE
01428	39	01465	00100	WA	CHECK SW 1
01440	38	01322	00100	WN	TYPE ROUTINE NO.
01452	47	01476	00300	BNI	TYPE TNS FIELD
01464	48	70707	2 0#	H	CHECK SW 3
01476	46	01344	00200	BI	HALT
01488	49	01536		B	CHECK SW 2

ROUTINE 003
CHECK TRANSFER NUMERIC STRIP
10 DIGITS + TO EVEN MEMORY

01500			X	WORKING AREA	
01512	0	#	X	WORKING AREA	
01524	98765	40321	X	COMPARE DATA	
01536	72	01179	01522	TNS	TRANSFER NUMERIC STRIP + FIELD
01548	24	01522	01535	C	CHECK FOR CORRECT RESULTS
01560	47	01608	01200	BNI	
01572	49	01668		B	
01584			X		
01596			X		
					ERROR ROUTINE
01608	46	01644	00100	BI	CHECK SW 1
01620	39	01657	00100	WA	TYPE ROUTINE NO.
01632	38	01513	00100	WN	TYPE TNS FIELD
01644	47	01668	00300	BNI	CHECK SW 3
01656	48	70707	3 0#	H	HALT
01668	46	01536	00200	BI	CHECK SW 2
01680	49	01728		B	

ROUTINE 004
CHECK TRANSFER NUMERIC STRIP
10 DIGITS - TO ODD MEMORY

01692			X	WORKING AREA	
01704	0		X	WORKING AREA	
01716	#	54321	06789	X	COMPARE DATA
01728	72	01199	01715	TNS	TRANSFER NUMERIC STRIP - FIELD
01740	24	01715	01727	C	CHECK FOR CORRECT RESULTS
01752	47	01800	01200	BNI	CHECK FOR E/Z
01764	49	01860		B	
01776			X		
01788			X		
					ERROR ROUTINE
01800	46	01836	00100	BI	CHECK SW 1
01812	39	01849	00100	WA	TYPE ROUTINE NO.
01824	38	01706	00100	WN	TYPE TNS FIELD
01836	47	01860	00300	BNI	CHECK SW 3
01848	48	70707	4 0#	H	HALT
01860	46	01728	00200	BI	CHECK SW 2
01872	49	01920		B	

ROUTINE 005
CHECK TRANSFER NUMERIC STRIP
10 DIGITS - TO EVEN MEMORY

01884			X	WORKING AREA
01896	0	#	X	WORKING AREA
01908		67890	12340	X COMPARE DATA
01920	72	01219	01906	TNS TRANSFER NUMERIC STRIP - FIELD
01932	24	01906	01919	C CHECK FOR CORRECT RESULTS
01944	47	01992	01200	BNI CHECK FOR E/Z
01956	49	02052		B
01968			X	
01980			X	
ERROR ROUTINE				
01992	46	02028	00100	BI CHECK SW 1
02004	39	02041	00100	WA TYPE ROUTINE NO.
02016	38	01897	00100	WN TYPE TNS FIELD
02028	47	02052	00300	BNI CHECK SW 3
02040	48	70707	5 0#	H HALT
02052	46	01920	00200	BI CHECK SW 2
02064	49	02136		B

ROUTINE 006
CHECK TRANSFER NUMERIC FILL
10 DIGITS + FROM ODD MEMORY

02076		0	X	WORKING AREA
02088			X	WORKING AREA
02100	#	12345	09876	X NUMERIC FIELD
02112		0717	27374	X COMPARE DATA
02124	75	70797	87776	X COMPARE DATA
02136	73	02099	02111	TNF TRANSFER NUMERIC FILL + FIELD
02148	44	02172	02080	BNF CHECK FOR NO FLAG
02160	49	02340		B BRANCH TO ERROR ROUTINE
02172	24	02099	02135	C CHECK FOR CORRECT RESULTS
02184	47	02208	01200	BNI CHECK FOR E/Z
02196	49	02268		B
ERROR ROUTINE				
02208	46	02244	00100	BI CHECK SW 1
02220	39	02257	00100	WA TYPE ROUTINE NO.
02232	38	02080	00100	WN TYPE TNF FIELD
02244	47	02268	00300	BNI CHECK SW 3
02256	48	70707	6 0#	H HALT
02268	46	02136	00200	BI CHECK SW 2
02280	49	02448		B

02292	49	55624	55963	X	INSERT
02304	45	44 4	65341	X	ED FLA
02316	47	57	4649	X	G P FI
02328	45	53440	3 0#	X	ELD. #
02340	46	02376	00100	BI	CHECK SW 1
02352	39	02257	00100	WA	TYPE ROUTINE NO.
02364	39	02293	00100	WA	TYPE REMOVED FLAG P FIELD
02376	49	02172		B	BRANCH TO COMPARE OF RESULTS

ROUTINE 007
CHECK TRANSFER NUMERIC FILL
10 DIGITS + FROM EVEN MEMORY

02388		0		X	WORKING AREA
02400				X	WORKING AREA
02412	#6	78901	2345	X	NUMERIC FIELD
02424		0767	77879	X	COMPARE DATA
02436	70	71727	37475	X	COMPARE DATA
02448	73	02411	02422	TNF	TRANSFER NUMERIC FILL + FIELD
02460	44	02484	02392	BNF	CHECK FOR NO FLAG
02472	49	02676		B	BRANCH TO ERROR ROUTINE
02484	24	02411	02447	C	CHECK FOR CORRECT RESULTS
02496	47	02544	01200	BNI	CHECK FOR E/Z
02508	49	02604		B	
02520				X	
02532				X	
02544	46	02580	00100	BI	ERROR ROUTINE
02556	39	02593	00100	WA	CHECK SW 1
02568	38	02391	00100	WN	TYPE ROUTINE NO.
02580	47	02604	00300	BNI	TYPE TNF FIELD
02592	48	70707	7 0#	H	CHECK SW 3
02604	46	02448	00200	BI	HALT
02616	49	02784		B	CHECK SW 2
02628	49	55624	55963	X	INSERT
02640	45	44 4	65341	X	ED FLA
02652	47	57	4649	X	G P FI
02664	45	53440	3 0#	X	ELD. #
02676	46	02712	00100	BI	CHECK SW 1
02688	39	02593	00100	WA	TYPE ROUTINE NO.
02700	39	02629	00100	WA	TYPE INSERTED FLAG P FIELD
02712	49	02484		B	BRANCH TO COMPARE OF RESULTS

ROUTINE 008
CHECK TRANSFER NUMERIC FILL
10 DIGITS - FROM ODD MEMORY

02724		0		X	WORKING AREA
02736				X	WORKING AREA
02748	#	98765	01234	X	NUMERIC FIELD
02760		0797	87776	X	COMPARE DATA
02772	75	70717	27354	X	COMPARE DATA
02784	73	02747	02759	TNF	TRANSFER NUMERIC FILL - FIELD
02796	44	02820	02728	BNF	CHECK FOR NO FLAG
02808	49	02964		B	BRANCH TO ERROR ROUTINE
02820	24	02747	02783	C	CHECK FOR CORRECT RESULTS
02832	47	02868	01200	BNI	CHECK FOR E/Z
02844	49	02928		B	
02856				X	
					ERROR ROUTINE
02868	46	02904	00100	BI	CHECK FOR SW 1
02880	39	02917	00100	WA	TYPE ROUTINE NO.
02892	38	02728	00100	WN	TYPE TNF FIELD
02904	47	02928	00300	BNI	CHECK SW 3
02916	48	70707	8 0#	H	HALT
02928	46	02784	00200	BI	CHECK SW 2
02940	49	03072		B	
02952				X	
02964	46	03000	00100	BI	CHECK SW 1
02976	39	02917	00100	WA	TYPE ROUTINE NO.
02988	39	02293	00100	WA	TYPE INSERTED FLAG P FIELD
03000	49	02820		B	BRANCH TO COMPARE RESULT.

ROUTINE 009
CHECK TRANSFER NUMERIC FILL
10 DIGITS - FROM EVEN MEMORY

03012		0		X	WORKING AREA
03024				X	WORKING AREA
03036	#1	06273	8490	X	NUMERIC FIELD
03048		0717	07672	X	COMPARE DATA
03060	77	73787	47950	X	COMPARE DATA
03072	73	03035	03046	TNF	TRANSFER NUMERIC FILL - FIELD
03084	44	03108	03016	BNF	CHECK FOR NO FLAG
03096	49	03264		B	BRANCH TO ERROR ROUTINE
03108	24	03035	03071	C	CHECK FOR CORRECT RESULTS
03120	47	03168	01200	BNI	CHECK FOR E/Z
03132	49	03228		B	
03144				X	
03156				X	

					ERROR ROUTINE
03168	46	03204	00100	BI	CHECK SW 1
03180	39	03217	00100	WA	TYPE ROUTINE NO.
03192	38	03015	00100	WN	TYPE TNF FIELD
03204	47	03228	00300	BNI	CHECK SW 3
03216	48	70707	9 0#	H	HALT
03228	46	03072	00200	BI	CHECK SW 2
03240	49	03324		B	
03252				X	
03264	47	03300	00100	BI	CHECK SW 1
03276	39	03217	00100	WA	TYPE ROUTINE NO.
03288	39	02629	00100	WA	TYPE INSERTED FLAG P FIELD
03300	49	03108		B	BRANCH TO COMPARE OF RESULT

ROUTINE 010
CHECK MOVE FLAG - FLAG TO NO FLAG

					WORKING AREA
03312		76	# 44#	X	
03324	15	03318	6	TQM	RESTORE P WORKING AREA TO ORIGINAL CONDITION
03336	15	03322	4	TQM	RESTORE Q WORKING AREA TO ORIGINAL CONDITION
03348	71	03318	03322	MR	MOVE FLAG
03360	14	03318	76	CM	CHECK FLAG MOVED TO P FIELD
03372	47	03444	01200	BNI	CHECK FOR E/Z
03384	14	03322	44	CM	CHECK FLAG REMOVED FROM Q FIELD
03396	47	03528	01200	BNI	CHECK FOR E/Z
03408	49	03588		B	
03420				X	
03432				X	
					ERROR ROUTINE
03444	46	03480	00100	BI	CHECK SW 1
03456	39	03493	00100	WA	TYPE ROUTINE NO.
03468	38	03317	00100	WN	TYPE P FIELD
03480	47	03504	00300	BNI	CHECK SW 3
03492	48	70717	0570#	H	HALT
03504	46	03324	00200	BI	CHECK SW 2
03516	49	03384		B	BRANCH TO Q FLAG CHECK
03528	46	03564	00100	BI	CHECK SW 1
03540	39	03577	00100	WA	TYPE ROUTINE NO.
03552	38	03321	00100	WN	TYPE Q FIELD
03564	47	03588	00300	BNI	CHECK SW 3
03576	48	70717	0580#	H	
03588	46	03324	00200	BI	CHECK SW 2
03600	49	03648		B	
03612				X	
03624				X	

ROUTINE 011
MOVE FLAG - FLAG TO FLAG

03636		75	# 11#	X	WORKING AREA
03648	15	03642	5	TDM	RESTORE P WORKING AREA TO ORIGINAL CONDITION
03660	15	03646	1	TDM	RESTORE Q WORKING AREA TO ORIGINAL CONDITION
03672	71	03642	03646	MF	MOVE FLAG
03684	14	03642	75	CM	CHECK FLAG REMAINED IN P FIELD
03696	47	03768	01200	BNI	CHECK FOR E/Z
03708	14	03646	11	CM	CHECK FLAG REMOVED FROM Q FIELD
03720	47	03852	01200	BNI	CHECK FOR E/Z
03732	49	03912		B	
03744				X	
03756				X	
ERROR ROUTINE					
03768	46	03804	00100	BI	CHECK SW 1
03780	39	03817	00100	WA	TYPE ROUTINE NO.
03792	38	03641	00100	WN	TYPE P FIELD
03804	47	03828	00300	BNI	CHECK SW 3
03816	48	70717	1570#	H	HALT
03828	46	03648	00200	BI	CHECK SW 2
03840	49	03708		B	BRANCH TO Q FLAG CHECK
03852	46	03888	00100	BI	CHECK SW 1
03864	39	03901	00100	WA	TYPE ROUTINE NO.
03876	38	03645	00100	WN	TYPE Q FIELD
03888	47	03912	00300	BNI	CHECK SW 3
03900	48	70717	1580#	H	HALT
03912	46	03648	00200	BI	CHECK SW 2
03924	49	03972		B	
03936				X	
03948				X	

ROUTINE 012
MOVE FLAG - NO FLAG TO FLAG

03960		79	# 33#	X	WORKING AREA
03972	15	03966	9	TDM	RESTORE P WORKING AREA TO ORIGINAL CONDITION
03984	15	03970	3	TDM	RESTORE Q WORKING AREA TO ORIGINAL CONDITION
03996	71	03966	03970	MF	MOVE FLAG
04008	14	03966	79	CM	CHECK FLAG REMOVED FROM P FIELD
04020	47	04080	01200	BNI	CHECK FOR E/Z
04032	14	03970	33	CM	CHECK Q FIELD REMAINED THE SAME
04044	47	04164	01200	BNI	CHECK FOR E/Z
04056	49	04224		B	
04068				X	

					ERROR ROUTINE
04080	46	04116	00100	BI	CHECK SW 1
04092	39	04129	00100	WA	TYPE ROUTINE NO.
04104	38	03965	00100	WN	TYPE P FIELD
04116	47	04140	00300	BNI	CHECK SW 3
04128	48	70717	2570#	H	HALT
04140	46	03972	00200	BI	CHECK SW 2
04152	49	04032		B	BRANCH TO Q FLAG CHECK
04164	46	04200	00100	BI	CHECK SW 1
04176	39	04213	00100	WA	TYPE ROUTINE NO.
04188	38	03969	00100	WN	TYPE Q FIELD
04200	47	04224	00300	BNI	CHECK SW 3
04212	48	70717	2580#	H	HALT
04224	46	03972	00200	BI	CHECK SW 2
04236	49	04284		B	
04248				X	
04260				X	

ROUTINE 013
MOVE FLAG - NO FLAG TO NO FLAG

					WORKING AREA
04272		78	# 22#	X	WORKING AREA
04284	15	04278	8	TDM	RESTORE P WORKING AREA TO ORIGINAL CONDITION
04296	15	04282	2	TDM	RESTORE Q WORKING AREA TO ORIGINAL CONDITION
04308	71	04278	04282	MF	MOVE FLAG - NO FLAG TO NO FLAG
04320	14	04278	78	CM	CHECK FOR NO FLAG IN P FIELD
04332	47	04404	01200	BNI	CHECK FOR E/Z
04344	14	04282	22	CM	CHECK FOR NO FLAG IN Q FIELD
04356	47	04488	01200	BNI	CHECK FOR E/Z
04368	49	04548		B	
04380					
04392					
					ERROR ROUTINE
04404	46	04440	00100	BI	CHECK SW 1
04416	39	04453	00100	WA	TYPE ROUTINE NO.
04428	38	04277	00100	WN	TYPE P FIELD
04440	47	04464	00300	BNI	CHECK SW 3
04452	48	70717	3570#	H	HALT
04464	46	04284	00200	BI	CHECK SW 2
04476	49	04344		B	BRANCH TO Q FLAG CHECK
04488	46	04524	00100	BI	CHECK SW 1
04500	39	04537	00100	WA	TYPE ROUTINE NO.
04512	38	04281	00100	WN	TYPE Q FIELD
04524	47	04548	00300	BNI	CHECK SW 3
04536	48	70717	3580#	H	HALT
04548	46	04284	00200	BI	CHECK SW 2
04560	49	04932		B	
04572				X	
04584				X	

ROUTINE 016

Check Transfer Numeric Strip
9 Digits v to Even Memory

04896			X	WORKING AREA
04908	0	#	X	WORKING AREA
04920	2345	67890	X	COMPARE DATA
04932	72	01237	04918	TNS TRANSFER NUMERIC STRIP - FIELD
04944	24	04918	04931	C COMPARE FOR CORRECT RESULTS
04956	47	05004	01200	BNI BRANCH TO ERROR ROUTINE
04968	49	05064		B
04980			X	
04992			X	

ERROR ROUTINE

05004	46	05040	00100	BI	CHECK SW1
05016	39	05053	00100	WA	TYPE ROUTINE NO.
05028	38	04910	00100	WN	TYPE COMPARE DATA
05040	47	05064	00300	BNI	CHECK SW 3
05052	48	70717	6 0#	H	HALT
05064	46	04932	00200	BI	CHECK SW 2
05076	49	04608		B	
05088				X	
05100				X	
05112				X	
05124		E		X	

ROUTINE 014
REPEAT ROUTINE

04596			000	X	WORKING AREA
04608	46	04620	01400	BI	TURN OFF OVERFLOW
04620	11	04607	01	AM	ADD ONE TO REPEAT CONSTANT
04632	47	01344	01400	BNI	CHECK FOR OVERFLOW
04644	34		00102	K	
04656	49	04836		B	

ROUTINE 015
TEST COMPLETED ROUTINE

04668	63	45626	3 43	X	TEST C
04680	56	54575	34563	X	OMPLET
04692	45	4403	4946	X	ED. IF
04704		6266	71	X	SW I
04716	56	4646	4155	X	OFF AN
04728	44	555	6 59	X	D NO R
04740	56	64634	95545	X	OUTINE
04752		55566	2 63	X	NOS T
04764	68	57454	423	X	YPED,
04776	63	45626	3 57	X	TEST P
04788	45	59465	65954	X	ERFORM
04800	45	44 5	75956	X	ED PRO
04812	57	45595	36803	X	PERLY.
04824	0#			X	#
04836	39	04669	00100	WA	TYPE TESTS COMPLETED
04848	46	01344	00400	BI	CHECK SW 4
04860					HALT
04872					
04884				X	
10000	12	34567	89098	X	
10001	76	54032	INMLK	X	
10002	J0	OPQR0	12345	X	
10003	OP	Q-234	56789	X	
10004	+L				

FP01– Floating Point Diagnostic


```
//=====
//  
// XX02 - Floating Point Diagnostic  
//  
// Program Switch settings:  
//  
// PS1:   ON - Bypass error type out  
//         OFF - Type out routine number on error  
// PS2:   ON - Loop in routine  
//         OFF - Continue to next routine  
// PS3:   ON - Stop on error  
//         OFF - Do not stop on error, continue  
// PS4:   not used  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY    - STOP  
// I/O       - STOP  
// O'FLOW    - PROGRAM  
//  
// Start addresses:  
//  
// 00402 - Full test  
//  
// Directions:  
//  
// 1. Load XX02_Floating_Point diagnostic  
// 2. Press START  
//=====
```


Sample Output - FP01

To be supplied when the
IBM 1620 Jr Simulator
supports floating point instructions.

----- 1: fp10-main.sps -----

```
1:1          * ****
1:2          * IBM 1620 FLOATING POINT PACKAGE TEST
1:3          * ****
1:4          *
1:5          *
1:6          * -----
1:7          * RUNNING IN 1620 SIMH
1:8          *      SET CPU FP
1:9          *      SET CPU DIV
1:10         *      SET CPU IA
1:11         *      SET CPU MOD1
1:12         *      SET CPU 60K
1:13         *
1:14         *      TURN ARITHMETIC CHECK STOP SWITCH OFF
1:15         *      DEPOSIT ARSTOP 0
1:16         *
1:17         *      SENSE SWITCHES OFF
1:18         *      DEPOSIT SW1 0
1:19         *      DEPOSIT SW2 0
1:20         *      DEPOSIT SW3 0
1:21         *      DEPOSIT SW4 0
1:22         *
1:23         *
1:24         * -----
1:25         *      SENSE SWITCH SETTINGS
1:26         *      SW1 ON = DO NOT PRINT ERROR
1:27         *      SW2 ON = REPEAT TEST
1:28         *      SW3 ON = HALT ON ERROR
1:29         *
1:30         *
1:31         * -----
1:32         *      REF 1. 227-5630-1 IBM 1620 FLOATING POINT FEATURE CE MANUAL
1:33         *      2. G26-5595-0 AUTOMATIC FLOATING POINT OPERATIONS SEPC
1:34         *
1:35         *      P = P ADDRESS
1:36         *      Q = Q ADDRESS
1:37         *      PF (MP) = CONTENTS OF P
1:38         *      QF (MQ) = CONTENTS OF Q
1:39         *      PE (EP) = EXPONENT OF THE FIELD AT THE P ADDRESS
1:40         *      QE (EQ) = EXPONENT OF THE FIELD AT THE Q ADDRESS
1:41         *      L = NUMBER OF DIGITS IN THE MANTISSA
```

```

1:41      * D = PE-QE (EP-EQ)
1:42      *
1:43      * REF 2 USES MP, MQ, AND EP, EQ
1:44      *
1:45      * =====
1:46      * SUMMARY
1:47      * DATA
1:48      *      FLOATING SHIFT RIGHT (FSR-08)
1:49      *      FLOATING SHIFT LEFT (FSL-05)
1:50      *      TRANSMIT FLOATING (TFL-06)
1:51      *
1:52      *      BRANCH AND TRANSMIT FLOATING (BTFL-07)
1:53      *
1:54      * ARITHMETIC - FROM REF 1 FLOATING POINT ARITHMETIC EXAMPLES
1:55      *          FLOATING POINT ARITHMETIC COMMAND RESULTS
1:56      *          ARE VERIFIED WITH FIXED POINT COMMANDS
1:57      * -----
1:58      *      FLOATING ADD (FADD-01)
1:59      *      EXAMPLES 1-5 EXPONENT CONFIGURATION
1:60      *      EXAMPLES 6-11 FRACTION (MANTISSA) CONFIGURATION
1:61      *
1:62      *      EXAMPLE 1 PE = QE / D = 0
1:63      *      EXAMPLE 2 (PE GT QE) SCAN Q -- FRACTION (MANTISSA) A
1:64      *          QF SHIFTED D (PE-QE) RIGHT TO ALIGN DECIM
1:65      *      EXAMPLE 3 (PE GT QE) SCAN Q -- FP EXIT -- D GT L
1:66      *          QE IS TOO SMALL-- Q DISCARDED-RESULT IS P
1:67      *      EXAMPLE 4 (PE LT QE) EXP TRANSMIT SCAN P - FRACTION
1:68      *          D LT L - PF SHIFTED D (PE-QE) RIGHT TO ALIG
1:69      *      EXAMPLE 5 (PE LT QE) EXP TRANSMIT SCAN P -- FP EXIT -
1:70      *          PE IS TOO SMALL-- P DISCARDED-RESULT IS Q
1:71      *      EXAMPLE 6 NO NORMALIZING REQUIRED---NO CARRY OUT
1:72      *      EXAMPLE 7 NORMALIZING SHIFT RIGHT REQUIRED -- CARRY O
1:73      *          EXPONENT ADJUST
1:74      *      EXAMPLE 8 NORMALIZING SHIFT RIGHT REQUIRED -- MACHINE
1:75      *          9...9 99 - CARRY OUT CAUSES EXPONENT OVERFI
1:76      *      EXAMPLE 9 ZERO FRACTION (MANTISSA) RESULT -- MACHINE
1:77      *      EXAMPLE 10 NORMALIZING SHIFT LEFT -- SHIFT LEFT -- ZE
1:78      *          EXPONENT ADJUST
1:79      *      EXAMPLE 11 NORMALIZING SHIFT LEFT -- EXPONENT OVERFL
1:80      *          MACHINE ZERO
1:81      *
1:82      *      FLOATING SUBTRACT (FSUB-02)
1:83      *          SAME AS FLOATING ADD EXCEPT THAT SIGN CONTROL
1:84      *          PROCEDURES FOR MQ ARE REVERSED.
1:85      *

```

```

1:86          *      FLOATING MULTIPLY (FMUL-03)
1:87          *      EXAMPLE 1. NO EXPONENT MODIFY REQUIRED
1:88          *      EXAMPLE 2. EXPONENT MODIFY REQUIRED
1:89          *      EXAMPLE 3. DIGIT FORCE -- MACHINE INFINITY
1:90          *      EXAMPLE 4. EXPONENT OVERFLOW -- MODIFY PE TO 99
1:91          *      EXAMPLE 5. SPECIAL CASE 999 X 199
1:92          *
1:93          *      FLOATING DIVIDE (FDIV-09)
1:94          *
1:95          *
1:96          * =====
1:97
1:98      00402 41 00000 00000  FPMAIN NOP    00000, 00000
1:99
1:100     00414 34 00000 00102      RCTY
1:101     00426 34 00000 00102      RCTY
1:102     00438 39 10319 00100      WATY  TITLE
1:103     00450 34 00000 00102      RCTY
1:104          * =====
1:105          *      RUN ALL TESTS
1:106          *
1:107          *
1:108          *ALL TESTS
1:109          * =====
1:110
1:111     00462 41 00000 00000  FIRST NOP
1:112

1:113     00474 17 08614 00486      BTM    SETS1, *+12,,      SET SW SUBROUTINE
1:114
1:115

1:116     00486 17 01236 00600      BTM    TFTST, *+114,,      TFL TEST
1:117     00499 06                  DC     2, 06,,,           TFL INSTRUCTION
1:118     00501 333333..33330@      DAC    50,===== TRANSMIT FLOATING (TFL 06) TESTS =====
1:119
1:120

1:121     00600 17 01236 00714      BTM    TFTST, *+114,,      FSL TEST
1:122     00613 05                  DC     2, 05,,,           FSL INSTRUCTION
1:123     00615 333333..33330@      DAC    50,===== FLOATING SHIFT LEFT (FSL 05) TESTS =====

```

1:124
 1:125
 1:126 00714 17 01236 $\bar{0}$ 0828 BTM TFTST, *+114,, FSR TEST
 1:127 00727 $\bar{0}$ 8 DC 2, 08,,, FSR INSTRUCTION
 1:128 00729 $\bar{3}$ 33333..33330@ DAC 50,===== FLOATING SHIFT RIGHT (FSR 08) TESTS ===
 1:129
 1:130
 1:131 00828 17 01236 $\bar{0}$ 0942 BTM TFTST, *+114,, BTFL TEST
 1:132 00841 $\bar{0}$ 7 DC 2, 07,,, BTFL INSTRUCTION
 1:133 00843 $\bar{3}$ 33333..33330@ DAC 50,==== BRANCH AND TRANSMIT FLOAT (BTFL 07) TESTS ==
 1:134
 1:135
 1:136
 1:137 00942 17 02466 $\bar{0}$ 1038 BTM FATST, *+96,, FADD TEST
 1:138 00955 $\bar{0}$ 1 DC 2, 01,,, FADD INSTRUCTION
 1:139 00957 $\bar{2}$ 1 DC 2, 21,,, ADD INSTRUCTION
 1:140 00959 $\bar{3}$ 33333..33330@ DAC 40,===== FLOATING ADD (FADD) TESTS =====@
 1:141
 1:142
 1:143
 1:144 01038 17 02466 $\bar{0}$ 1134 BTM FATST, *+96,, FSUB TEST
 1:145 01051 $\bar{0}$ 2 DC 2, 02,,, FSUB INSTRUCTION
 1:146 01053 $\bar{2}$ 2 DC 2, 22,,, SUBTRACT INSTRUCTION
 1:147 01055 $\bar{3}$ 33333..33330@ DAC 40,===== FLOATING SUB (FSUB) TESTS =====@
 1:148
 1:149
 1:150 01134 17 06372 $\bar{0}$ 1146 BTM FMTST, *+12,, FMUL TEST
 1:151
 1:152 * ----- PRINT END OF TESTING MESSAGE AND HALT
 1:153

```

1:154 01146 41 00000 00000 ALDON NOP
1:155 01158 34 00000 00102 RCTY
1:156 01170 34 00000 00102 RCTY
1:157 01182 39 10389 00100 WATY DONE
1:158 01194 34 00000 00102 RCTY
1:159 01206 48 00000 00000 H 00000, 00000,, END OF PROGRAM
1:160 01218 48 00000 00000 H 00000, 00000,, WILL SHOW (H) AS NEXT INSTRUCTI
1:161
1:162
1:163 * =====
1:164 * TEST SUBROUTINES FOLLOW
1:165 * =====
1:166 * =====
1:167 * =====
1:168

```

----- 2: fp20-tftest.sps -----

```

2:1 * =====
2:2 * TFL, BTFL, FSL, FSR TESTS
2:3 *
2:4 *TFTST
2:5 * =====
2:6 01234 00005 DS 5,,, RETURN ADDRESS FROM BTM
2:7 01236 41 00000 00000 TFTST NOP 00000, 00000
2:8
2:9 * INSTRUCTION ARG
2:10 01248 26 01283 01235 TF *+35, TFTST-1,,,
2:11 01260 12 01283 00101 SM *+23, 101,,,
2:12 01272 26 01290 00000 TF TFINS+6, 0,, TO FLOATING POINT INSTRUCTION
2:13 01284 41 00000 00000 TFINS NOP 00000,00000,, KEEP INSTRUCTION ARG IN P ADDRESS
2:14
2:15 * TEST TITLE ARG
2:16 01296 26 01331 01235 TF *+35, TFTST-1,,,
2:17 01308 12 01331 00099 SM *+23, 99,,,
2:18 01320 16 01362 00000 TFM TFWT+6, 0,, TO WRITE TITLE
2:19
2:20 * TITLE
2:21
2:22 01332 34 00000 00102 RCTY
2:23 01344 34 00000 00102 RCTY

```

-

2:24 01356 39 00000 00100 TFWT WATY 00000,,6, MAIN TEST TITLE
2:25 01368 34 00000 00102 RCTY
2:26
2:27
2:28 * INITIALIZE TABLE POINTER
2:29

2:30 01380 16 10767 00000 TFM TFINC, 0,, TABLE POINTER
2:31
2:32 * POINT TO TEST DATA TABLES
2:33 * RESET

2:34 01392 16 10782 10802 TFM TFTBL, TFTBI
2:35 01404 16 10787 10837 TFM TFTBL+5, TFTBD
2:36 01416 26 10777 10782 TF TFTBP, TFTBL
2:37 01428 26 10782 10787 TF TFTBP+5, TFTBL+5
2:38
2:39
2:40 01440 41 00000 00000 TFNXT NOP
2:41
2:42 * ADVANCE TO NEXT ADDRESS IN TABLES
2:43
2:44 01452 21 10777 10767 A TFTBP, TFINC,, ID MESSAGE TABLE
2:45 01464 14 10777 00000 CM TFTBP, 00000, 6, RETURN ON ZERO ADDRESS
2:46 01476 46 01235 01200 BE TFTST-1,,6, RETURN
2:47
2:48 01488 21 10782 10767 A TFTBP+5, TFINC,, DATA TABLE
2:49

2:50 01500 16 10767 00005 TFM TFINC, 5
2:51

2:52 01512 26 10792 10777 TF TFI, TFTBP,11
2:53 01524 26 10797 10782 TF TFD, TFTBP+5,11
2:54
2:55 * START OF CURRENT TEST
2:56 01536 34 00000 00102 TFRPT RCTY

2:57 01548 39 10792 00100 WATY TFI,,6, TEST TITLE
2:58 01560 34 00000 00101 SPTY

2:60 * FP INSTRUCTION TEST FOLLOW.
 2:61 * TEST DATA FIELD IN ADDRESS REFERENCED
 2:62 * BY TFD ASSUMED TO BE A 100 DIGIT MANTISSA
 2:63
 2:64 * SELECT INSTRUCTION (TFINS)
 2:65 01572 25 14955 14953 TD STAT, FALSE
 2:66 01584 26 10658 10554 TF WK1, WK1Z,, ZERO WK1
 2:67 01596 33 10558 00000 CF WK1-100
 2:68 01608 26 10761 10554 TF WK2, WK1Z,, ZERO WK2
 2:69 01620 33 10661 00000 CF WK2-100
 2:70
 2:71 01632 26 02195 10797 TF TEXCK+11, TFD,, EXP COMPARE
 2:72
 2:73 * WHEN TFL(06)
 2:74 01644 14 01290 00006 CM TFINS+6, 06, 10,, TFL
 2:75 01656 47 01740 01200 BNE TFSL
 2:76 01668 06 10658 10797 TFL WK1, TFD, 11
 2:77 01680 26 01715 10797 TF *+35, TFD,, MANTISSA COMPARE
 2:78 01692 12 01715 00002 SM *+23, 2
 2:79 01704 24 10656 00000 C WK1-2, 00000,, MANTISSA
 2:80 01716 46 02184 01200 BE TEXCK,,, EXPONENT
 2:81 01728 49 02220 00000 B TER
 2:82
 2:83 * WHEN FSL(05)
 2:84 * FSL LEAVES SIGN FLAG IN ORIGINAL POSITION
 2:85 * THE COMPARE (C) COMPARES THE SHIFTED DATA
 2:86 * TO THE ORIGINAL. IF THE ORIGINAL IS NEGATIVE
 2:87 * THEN TO MAKE THE COMPARE WORK THE SHIFTED DATA
 2:88 * NEEDS A SIGN (FLAG LOW ORDER)
 2:89 01740 14 01290 00005 TFSL CM TFINS+6, 05, 10,, FSL
 2:90 01752 47 01896 01200 BNE TFSR
 2:91 01764 26 01799 10797 TF *+35, TFD
 2:92 01776 12 01799 00101 SM *+23, 101
 2:93 01788 31 10557 00000 TR WK1-101, 00000,, COPY FP NUMBER TO WORK
 2:94 01800 05 10578 10656 FSL WK1-80, WK1-2
 2:95 01812 26 01871 10797 TF TFSLC+11, TFD
 2:96 01824 12 01871 00002 SM TFSLC+11, 2
 2:97 01836 44 01860 01871 BNF TFSLC, TFSLC+11, 11

2:98 01848 32 10627 00000 SF WK1-31
 2:99 01860 24 10627 00000 TFSLC C WK1-31,, 00000,, MANTISSA
 2:100 01872 46 02184 01200 BE TEXCK,,, EXPONENT
 2:101 01884 49 02220 00000 B TER
 2:102
 2:103 * WHEN FSR(08)
 2:104 * FIRST MAKE LEFT SHIFTED COPY TO WK1 AND CORRECT SIGN
 2:105 01896 14 01290 00008 TFSR CM TFINS+6, 08, 10,, FSR
 2:106 01908 47 02112 01200 BNE TBTFL
 2:107 01920 26 01955 10797 TF *+35, TFD
 2:108 01932 12 01955 00002 SM *+23, 2
 2:109 01944 26 10608 00000 TF WK1-50, 00000,, COPY FP NUMBER TO WORK
 2:110 01956 26 01979 10797 TF *+23, TFD
 2:111 01968 26 10658 00000 TF WK1, 00000,, AND EXPONENT
 2:112 01980 26 02015 10797 TF *+35, TFD
 2:113 01992 12 02015 00002 SM *+23, 2
 2:114 02004 44 02040 00000 BNF TFSRF, 00000
 2:115 02016 33 10608 00000 CF WK1-50
 2:116 02028 32 10656 00000 SF WK1-2
 2:117 02040 08 10656 10608 TFSRF FSR WK1-2, WK1-50
 2:118 02052 26 02087 10797 TF TFSRC+11, TFD
 2:119 02064 12 02087 00002 SM TFSRC+11, 2
 2:120 02076 24 10656 00000 TFSRC C WK1-2, 00000,, MANTISSA
 2:121 02088 46 02184 01200 BE TEXCK,,, EXPONENT
 2:122 02100 49 02220 00000 B TER
 2:123
 2:124
 2:125 * WHEN BTFL(07)
 2:126 02112 26 02183 10797 TBTFL TF *+71, TFD
 2:127 02124 26 02159 10797 TF *+35, TFD
 2:128 02136 12 02159 00101 SM *+23, 101
 2:129 02148 31 10557 00000 TR WK1-101, 00000,, COPY FP NUMBER TO WORK
 2:130 02160 16 02460 02184 TFM RETAD, *+24,,
 2:131 02172 07 02392 00000 BTFL BTSUB, 00000,, BTFL RETURN
 2:132 USES WK1 AND SETS STAT
 2:133 02184 24 10658 00000 TEXCK C WK1, 00000,, EXPONENT
 2:134 02196 47 02220 01200 BNE TER
 2:135 02208 25 14955 14952 TD STAT, TRUE
 2:136

2:137
2:138
2:139 * CHECK COMPARE STATUS
2:140 * CALL ERROR ROUTINE AND RETURN ITS STATUS IN ESTAT

2:141 02220 17 09518 02242 TER BTM ERTN, *+22,, TRANSMIT ADDRESS OF NEXT INSTRUCTION

2:142 02236 1495514954 DSA STAT, ESTAT
2:143
2:144 02242 43 02266 14954 BD TFEND, ESTAT
2:145 02254 49 01536 00000 B TFRPT,,, RERUN TEST
2:146
2:147 02266 41 00000 00000 TFEND NOP 00000, 00000
2:148
2:149 02278 49 01440 00000 B TFXNXT,,, GOTO NEXT TEST
2:150
2:151

----- 3: fp21-tftest.sps -----

3:1 * -----
3:2 * SUBROUTINE STATUS = BTSUB(FP1, WK1)
3:3 * FP1 IS COPIED TO WK1
3:4 * BRANCH AND TRANSMIT COPIES WK1 TO SUBROUTINE-BTSUB
3:5 * BTSUB GETS WK1 CONTENTS JUST BEFORE BTSUB ADDRESS (BTSUB-1)
3:6 * BTSUB CALLS CF -COMPARE FLOATING SUBROUTINE TO
3:7 * COMPARE WK1 TO ORIGINAL FP1
3:8 * CF RETURNS COMPARE STATUS IN STAT
3:9
3:10 *
3:11 *
3:12 *BTSUB
3:13 * -----
3:14

3:15 02339 000000..000000 DC 50, 0
3:16 02340 000000..000000 DSC 50, 0
3:17 02390 00 DSC 2, 0
3:18 02392 41 00000 00000 BTSUB NOP
3:19
3:20
3:21 *COMPARE MANTISSA AND EXPONENT SEPARATELY
3:22 * WITH SUBROUTINE CF (COMPARE FLOATING)

3:23 02404 17 09922 02432 BTM CF, *+28,, TRANSMIT RETURN ADDRESS

```

3:24    02420 023911..814955      DSA    BTSUB-1, WK1, STAT
3:25
3:26    02432 41 00000 00000     NOP    00000, 00000,,      RETURN TO HERE
3:27
3:28          * RETURN
3:29          * NOTE BB WILL NOT WORK HERE,
3:30          * BECAUSE THIS SUBROUTINE CONTAINS A BTM

3:31    02444 49 02460 00000     B      RETAD,,6
3:32
3:33          * -----
3:34          * SUBROUTINE BTFL RETURN ADDRESS
3:35    02460 00005     RETAD DS 5
3:36
3:37

```

----- 4: fp30-fadd.sps -----

```

4:1          * =====
4:2          * FADD TEST
4:3          *
4:4          *FADD
4:5          * =====
4:6    02465 00005     DS 5,,,           RETURN ADDRESS FROM BTM
4:7    02466 41 00000 00000   FATST NOP 00000, 00000
4:8
4:9          * GET ADD / SUBTRACT INST AND TITLE
4:10   02478 26 02513 02465     TF *+35, FATST-1,,,
4:11   02490 12 02513 00083     SM  *+23, 83,,,
4:12   02502 26 03043 00000     TF  FARIT+1, 0,,       TO FLOATING POINT INSTRUCTION
4:13
4:14   02514 26 02549 02465     TF  *+35, FATST-1,,,
4:15   02526 12 02549 00081     SM  *+23, 81,,,
4:16   02538 16 03118 00000     TFM  ADSA, 0,,       FIXED INST TO FIXED ADD/SUB SUB
4:17
4:18   02550 26 02585 02465     TF  *+35, FATST-1,,,
4:19   02562 12 02585 00079     SM  *+23, 79,,,
4:20   02574 16 02616 00000     TFM  WTITL+6, 0,,       TO WRITE TITLE
4:21

```

4:22 * TITLE
 4:23
 4:24 02586 34 00000 00102 RCTY
 4:25 02598 34 00000 00102 RCTY
 4:26 02610 39 00000 00100 WTITL WATY 00000,,6, MAIN FADD / FSUB TEST TITLE
 4:27 02622 34 00000 00102 RCTY
 4:28
 4:29
 4:30 * INITIALIZE TABLE POINTER
 4:31
 4:32 02634 16 11713 00000 TFM FAINC, 0,, TABLE POINTER
 4:33
 4:34 02646 14 03043 00002 CM FARIT+1, 02, 10, CHOOSE FADD / SUB TABLES
 4:35 02658 46 02754 01200 BE SUBTB
 4:36
 4:37 * POINT TO FADD TEST DATA TABLES
 4:38 * RESET
 4:39 02670 16 11733 11778 TFM FATBL, FATBI
 4:40 02682 16 11738 11833 TFM FATBL+5, FATBP
 4:41 02694 16 11743 11888 TFM FATBL+10, FATBQ
 4:42 02706 26 11728 11733 TF ASTBL, FATBL
 4:43 02718 26 11733 11738 TF ASTBL+5, FATBL+5
 4:44 02730 26 11738 11743 TF ASTBL+10, FATBL+10
 4:45 02742 49 02826 00000 B FANXT
 4:46
 4:47 * POINT TO FSUB TEST DATA TABLES
 4:48 * RESET
 4:49 02754 16 11748 11943 SUBTB TFM FSTBL, FSTBI
 4:50 02766 16 11753 11998 TFM FSTBL+5, FSTBP
 4:51 02778 16 11758 12053 TFM FSTBL+10, FSTBQ
 4:52 02790 26 11728 11748 TF ASTBL, FSTBL
 4:53 02802 26 11733 11753 TF ASTBL+5, FSTBL+5
 4:54 02814 26 11738 11758 TF ASTBL+10, FSTBL+10
 4:55
 4:56
 4:57 02826 41 00000 00000 FANXT NOP

4:58
 4:59 * ADVANCE TO NEXT ADDRESS IN TABLES
 4:60 02838 21 11728 11713 A ASTBL, FAINC,, ID MESSAGE TABLE
 4:61 02850 14 11728 00000 CM ASTBL, 00000, 6, RETURN ON ZERO ADDRESS
 4:62 02862 46 02465 01200 BE FATST-1,, 6, RETURN
 4:63
 4:64 02874 21 11733 11713 A ASTBL+5, FAINC,, P DATA TABLE
 4:65 02886 21 11738 11713 A ASTBL+10, FAINC,, Q DATA TABLE
 4:66
 4:67 02898 16 11713 00005 TFM FAINC, 5
 4:68
 4:69 02910 26 11763 11728 TF FAI, ASTBL, 11
 4:70 02922 26 11768 11733 TF FAP, ASTBL+5, 11
 4:71 02934 26 11773 11738 TF FAQ, ASTBL+10, 11
 4:72
 4:73 * START OF CURRENT FADD TEST
 4:74 02946 41 00000 00000 FARPT NOP
 4:75 02958 34 00000 00102 RCTY
 4:76 02970 39 11763 00100 WATY FAI,, 6, FADD TEST TITLE
 4:77 02982 34 00000 00101 SPTY
 4:78
 4:79 *---- NEED TO PRESERVE AUGEND - MOVE TO FASUM FIRST
 4:80 *---- AVOID TFL FOR FLOATING POINT TESTS
 4:81
 4:82 02994 26 12205 11768 TF FASUM, FAP, 11, P POINTS TO EXPONENT
 4:83 03006 12 11768 00002 SM FAP, 00002,, P NOW POINTS TO MANTISSA
 4:84 03018 26 12203 11768 TF FASUM-2, FAP, 11,
 4:85 03030 11 11768 00002 AM FAP, 00002,, P AGAIN POINTS TO EXPONENT
 4:86
 4:87 * FADD / FSUB INST ARE PASSED IN ARGS, WHICH REPLACE FADD BEI
 4:88
 4:89 03042 01 12205 11773 FARIT FADD FASUM, FAQ, 11, FLOAT ARITHMETIC
 4:90

```

4:91
4:92          * INDICATOR CHECK
4:93    03054 47 03078 01400      BNV   *+24
4:94    03066 39 15015 00100      WATY   OVM1,,,           OVERFLOW MESSAGE
4:95    03078 47 03102 01500      BNXV   *+24
4:96    03090 39 15037 00100      WATY   XCKM1,,,           EXPONENT OVERFLOW MESSAGE
4:97
4:98
4:99          * CALL SUBROUTINE A21
4:100          * A (21) MANTASSAS AND COMPUTE EXPONENTS

4:101    03102 17 03216 03140      BTM   A21,   *+38,,     TRANSMIT RETURN ADDRESS
4:102    03118 000001..514955  ADSA   DSA   00000, -FAP, -FAQ, FASUM, STAT
4:103
4:104
4:105          * CHECK COMPARE STATUS
4:106          * CALL ERROR ROUTINE AND RETURN ITS STATUS IN ESTAT

4:107    03140 17 09518 03162      BTM   ERTN, *+22,,     TRANSMIT ADDRESS OF NEXT INSTRUCTION
4:108    03156 1495514954      DSA   STAT, ESTAT
4:109
4:110    03162 43 03186 14954      BD    FAEND, ESTAT
4:111    03174 49 02946 00000      B     FARPT,,,     RERUN TEST
4:112
4:113    03186 41 00000 00000      FAEND NOP   00000, 00000
4:114
4:115    03198 49 02826 00000      B     FANXT,,,     GOTO NEXT FADD TEST
4:116

```

----- 5: fp31-fixadd.sps -----

```

5:1          * -----
5:2          * SUBROUTINE STATUS = A21(FXARI, FP1, FP2, FP-SUM)
5:3          *      COMPARES FADD SUM WITH A(21)/EX (EX ADJUST)
5:4          *      -- OR -- FSUB WITH S(22) PER FIRST ARGUMENT (FXARI)
5:5          *
5:6          * REF (1) 227-5630-1 IBM 1620 FLOATING POINT FEATURE CE MANUAL
5:7          *
5:8          * VERIFIES FADD RESULT BY USING A(21) TO ADD MANTISSAS
5:9          * AND SEPARATELY HANDLES EXPONENTS.
5:10         * FLOATING POINT NUMBER IS CREATED FROM A(21) SUM AND
5:11         * (ADJUSTED) EXPONENTS
5:12         *

```

```

5:13          * HANDLES CASES DESCRIBED IN REF (1) FLOATING ADD EXAMPLES
5:14          *
5:15          * EXAMPLE 1 PE = QE / D = 0 NO NORMALIZING - NO CARRY OUT
5:16          * EXAMPLE 2 PE GT QE - D LT L QF SHIFTED RT TO ALIGN DECIMAL
5:17          * EXAMPLE 3 PE GT QE - D GT L - Q DISCARDED-RESULT IS P
5:18          * EXAMPLE 4 PE LT QE- D LT L PF SHIFTED RT TO ALIGN DECIMAL P
5:19          * EXAMPLE 5 PE LT QE - D GT L P DISCARDED-RESULT IS Q
5:20          *
5:21          * EXAMPLE 6 NO NORMALIZING REQUIRED-NO CARRY OUT-SAME AS EXAMPLE
5:22          *
5:23          * EXAMPLE 7 NORMALIZING SHIFT RT - CARRY OUT- EXP ADJUST
5:24          * EXAMPLE 8 NORM SHIFT RT-CARRY OT CAUSES EXP OV MACH INFINITE
5:25          * EXAMPLE 9 ZERO FRACTION RESULT - MACHINE ZERO (0...0 -99)
5:26          * EXAMPLE 10 NORMALIZING SHIFT LEFT - SHIFT LEFT- EXP ADJ
5:27          * EXAMPLE 11 NORMALIZING SHIFT LEFT- EXP OV MACHINE ZERO (0.0000)
5:28          *
5:29          * USAGE
5:30          *      BTM    A21,   *+38,,           TRANSMIT RETURN ADDRESS
5:31          *      DSA    FXARI, -FAP, -FAQ, FASUM, STAT
5:32          *      NOP    00000, 00000,,       RETURN TO HERE
5:33          *
5:34          * FXARI = A OR S ARITHMETIC INSTRUCTION
5:35          * A21A1 = -FAP (INDIRECT ADDRESS) FLOATING POINT NUMBER
5:36          * A21A2 = -FAQ (INDIRECT ADDRESS) FLOATING POINT NUMBER
5:37          * A21D1 = FASUM (DATA FIELD) FLOATING POINT FADD SUM
5:38          * A21RC = STAT ( DATA, RETURN CODE )
5:39          * RETURNS STATUS OF COMPARE
5:40          *
5:41          *
5:42          *A21
5:43          * -----
5:44      03214 00005          DS    5,,           RETURN ADDRESS
5:45      03216 41 00000 00000  A21    NOP    00000, 00000
5:46
5:47
5:48          * GET FIRST ARGUMENT
5:49      03228 26 03263 03215  TF    *+35, A21-1,,,
5:50      03240 12 03263 00022  SM    *+23, 00022,,,
5:51      03252 26 03275 00000  TF    *+23, 0,,,
5:52      03264 26 04165 00000  TF    FXAR1+1, 0,, -- A / S FIXED POINT ARITH INSTRUC
5:53      03276 26 03299 03275  TF    *+23, *-1
5:54      03288 26 04789 00000  TF    FXAR2+1, 0,, -- A / S FIXED POINT ARITH INSTRUC
5:55
5:56

```

5:57 * GET SECOND ARGUMENT (INDIRECT)
 5:58 03300 26 03335 03215 TF *+35, A21-1,,

 5:59 03312 12 03335 00017 SM *+23, 00017,,
 5:60 03324 26 03359 00000 TF *+35, 0,,
 5:61 03336 33 03359 00000 CF *+23
 5:62 03348 26 06253 00000 TF A21A1, 0,,
 5:63
 5:64
 5:65 * GET THIRD ARGUMENT (INDIRECT)
 5:66 03360 26 03395 03215 TF *+35, A21-1,,

 5:67 03372 12 03395 00012 SM *+23, 00012,,
 5:68 03384 26 03419 00000 TF *+35, 0,,
 5:69 03396 33 03419 00000 CF *+23
 5:70 03408 26 06258 00000 TF A21A2, 0,,
 5:71
 5:72
 5:73 * GET FOURTH ARGUMENT
 5:74 03420 26 03455 03215 TF *+35, A21-1,,

 5:75 03432 12 03455 00007 SM *+23, 00007,,
 5:76 03444 26 03467 00000 TF *+23, 0,,
 5:77 03456 26 06360 00000 TF A21D1, 0,, -- GETS EXP
 5:78
 5:79 03468 26 03503 03467 TF *+35, *-1,,

 5:80 03480 12 03503 00002 SM *+23, 00002,
 5:81 03492 26 06358 00000 TF A21D1-2, 0,, -- GETS MANTISSA
 5:82
 5:83
 5:84 * EXPONENT
 5:85 * HANDLE RESULTANT EXPONENT BASED ON L AND D
 5:86 * AML = MANTISSA LENGTH AED = ABS (EP-EQ)
 5:87
 5:88 03504 26 05403 06042 TF AEP, AZ5,, CLEAR
 5:89 03516 26 05511 06042 TF AEQ, AZ5,, CLEAR

 5:90 03528 21 05403 06253 A AEP, A21A1, 11, EXPONENT P

 5:91 03540 21 05511 06258 A AEQ, A21A2, 11, EXPONENT Q

 5:92
 5:93
 5:94 * SUBTRACT MP AND MQ EXPONENTS AED = ABS (EP-EQ)
 5:95 03552 26 05281 06042 TF AED, AZ5,, CLEAR

5:96 03564 21 05281 05403 A AED, AEP,, EXPONENT P
 5:97 03576 22 05281 05511 S AED, AEQ,, EXPONENT Q
 5:98 03588 33 05281 00000 CF AED,,, CLEAR SIGN FLAG TO GET A
 5:99
 5:100
 5:101 * MANTISSA
 5:102 03600 12 06253 00002 SM A21A1, 00002
 5:103 03612 12 06258 00002 SM A21A2, 00002
 5:104 03624 26 05397 06243 TF AMP, AZ101
 5:105 03636 26 05505 06243 TF AMQ, AZ101
 5:106 03648 26 05397 06253 TF AMP, A21A1, 11, MANTISSA P
 5:107 03660 26 05505 06258 TF AMQ, A21A2, 11, MANTISSA Q
 5:108
 5:109
 5:110 * MANTISSA LENGTH OF FIRST ARG IN AML
 5:111 * L REPRESENTS MANTISSA LENGTH
 5:112 03672 16 05276 00001 TFM AML, 00001,
 5:113 03684 26 05296 06253 TF AMPA, A21A1,, AMPA HIGH ORDER ADDRESS C
 5:114
 5:115 03696 12 05296 00001 AL SM AMPA, 1
 5:116 03708 11 05276 00001 AM AML, 1
 5:117 03720 44 03696 05296 BNF AL, AMPA, 11
 5:118
 5:119
 5:120 * ALIGN DECIMAL POINTS
 5:121 03732 24 05403 05511 C AEP, AEQ
 5:122 03744 46 04104 01200 BE ADDM,,, NO ALIGN-ADD MANTISSAS UN
 5:123 03756 47 03780 01300 BL ADJP,,, PE LT QE
 5:124 03768 46 03936 01100 BH ADJQ,,, PE GT QE
 5:125
 5:126
 5:127 * ADJUST P MANTISSA
 5:128 * PE LT QE - D GT L - P DISCARDED-RESULT IS Q
 5:129 03780 24 05281 05276 ADJP C AED, AML,, COMPARE D TO L
 5:130 03792 47 03840 01300 BL ADJP1
 5:131 03804 26 05613 05505 TF AMPQ, AMQ
 5:132 03816 26 05619 05511 TF AEPQ, AEQ

5:133 03828 49 05148 00000 B AMKFP,,, GOTO MAKE FP NUMBER FROM Q ONLY
 5:134

5:135 03840 16 03875 05397 ADJP1 TFM *+35, AMP,, RESTORE ADDRESS IN TF BE
 5:136 03852 22 03875 05281 S *+23, AED,, ALIGN DECIMAL POINTS
 5:137 03864 26 05397 05397 TF AMP, AMP

5:138 03876 16 03918 05397 TFM *+42, AMP,, RESTORE ADDRESS IN MF BE
 5:139 03888 22 03918 05276 S *+30, AML

5:140 03900 11 03918 00001 AM *+18, 00001
 5:141 03912 71 05397 05397 MF AMP, AMP
 5:142 03924 49 04104 00000 B ADDM

5:143
 5:144
 5:145 * ADJUST Q MANTISSA
 5:146 * PE GT QE - D GT L - Q DISCARDED-RESULT IS P
 5:147 03936 41 00000 00000 ADJQ NOP 00000, 00000,,
 5:148 03948 24 05281 05276 C AED, AML,, COMPARE D TO L
 5:149 03960 47 04008 01300 BL ADJQ1
 5:150 03972 26 05613 05397 TF AMPQ, AMP
 5:151 03984 26 05619 05403 TF AEPQ, AEP
 5:152 03996 49 05148 00000 B AMKFP,,, GOTO MAKE FP NUMBER FROM P ONLY
 5:153

5:154 04008 16 04043 05505 ADJQ1 TFM *+35, AMQ,, RESTORE ADDRESS IN TF BE
 5:155 04020 22 04043 05281 S *+23, AED,, ALIGN DECIMAL POINTS
 5:156 04032 26 05505 05505 TF AMQ, AMQ

5:157 04044 16 04086 05505 TFM *+42, AMQ,, RESTORE ADDRESS IN MF BE
 5:158 04056 22 04086 05276 S *+30, AML

5:159 04068 11 04086 00001 AM *+18, 00001
 5:160 04080 71 05505 05505 MF AMQ, AMQ
 5:161 04092 49 04104 00000 B ADDM

5:162
 5:163
 5:164 * ADD (ADJUSTED) P TO Q MANTISSAS
 5:165 04104 41 00000 00000 ADDM NOP 00000, 00000,,
 5:166 04116 25 05622 14953 TD ANRM, FALSE,, CLEAR NORMALIZE FLAG
 5:167 04128 25 05621 14953 TD AOV, FALSE,, CLEAR OVERFLOW FLAG
 5:168 04140 26 05613 06243 TF AMPQ, AZ101,, CLEAR
 5:169 04152 26 05613 05397 TF AMPQ, AMP
 5:170 04164 21 05613 05505 FXAR1 A AMPQ, AMQ,, ARGs MAY REPLACE A WITH S
 5:171

5:172
 5:173 * CARRY-OUT OVERFLOW INDICATOR CHECK
 5:174 04176 47 04212 01400 BNV ASHFT
 5:175 04188 25 05621 14952 TD AOV, TRUE,, SET OVERFLOW FLAG
 5:176 04200 49 04548 00000 B AZCK
 5:177
 5:178
 5:179 * NORMALIZE MANTISSA IN AMPQ
 5:180 * SHIFT LEFT
 5:181 * COUNT LEADING ZEROS IN AZCNT
 5:182 * REQUIRES EXPONENT ADJUST
 5:183 04212 41 00000 00000 ASHFT NOP 00000, 00000
 5:184 04224 24 05613 06243 C AMPQ, AZ101,, ZERO MANTISSA CHECK
 5:185 04236 46 04572 01200 BE AZRO
 5:186
 5:187 04248 25 05622 14953 TD ANRM, FALSE,, CLEAR NORMALIZE FLAG
 5:188 04260 16 05291 00000 TFM AZCNT, 00000,, CLEAR LEADING ZERO COUNTER
 5:189
 5:190 04272 16 04319 05613 TFM *+47, AMPQ
 5:191 04284 22 04319 05276 S *+35, AML
 5:192 04296 11 04319 00001 AM *+23, 00001
 5:193 04308 43 04548 05613 BD AZCK, AMPQ,, LEADING ZERO CHECK
 5:194
 5:195 04320 26 06034 06243 TF AWK1, AZ101,, CLEAR WORK
 5:196 04332 25 05622 14952 TD ANRM, TRUE,, SET NORMALIZE FLAG
 5:197 04344 16 05286 00000 TFM AFLG, 00000
 5:198 04356 16 05296 05613 TFM AMPA, AMPQ,, POINTS HI MANTISSA ADDR
 5:199
 5:200 04368 12 05296 00001 AFLAG SM AMPA, 00001,, SEARCH FOR ORIGINAL FLAG
 5:201 04380 44 04368 05296 BNF AFLAG, AMPA, 11
 5:202 04392 33 05296 00000 CF AMPA, 00000, 6, CLEAR IT
 5:203
 5:204 04404 43 04464 05296 ADIGT BD ASF, AMPA, 11, 1ST DIGIT TO SHIFT LEFT
 5:205 04416 11 05291 00001 AM AZCNT, 00001,, COUNT LEADING ZEROS

5:206	04428 11 05286 00001		AM	AFLG, 00001	
5:207	04440 11 05296 00001		AM	AMPA, 00001	
5:208	04452 49 04404 00000		B	ADIGT	
5:209					
5:210	04464 16 04530 05613	ASF	TFM	ASF2+6, AMPQ,,	INITIALIZE TF P ADDRESS
5:211	04476 32 05296 00000		SF	AMPA, 00000, 6,	FLAG 1ST DIGIT
5:212	04488 26 06034 05613		TF	AWK1, AMPQ,,	NOW LEFT SHIFT MANTISSA
5:213	04500 26 05613 06243		TF	AMPQ, AZ101	
5:214	04512 22 04530 05286		S	*+18, AFLG	
5:215	04524 26 05613 06034	ASF2	TF	AMPQ, AWK1,,	SHIFT COMPLETE
5:216	04536 49 04824 00000		B	EXADJ,,,	GOTO EXPONENT ADJUST
5:217					
5:218					
5:219					
5:220				* ZERO CHECK GOTO MACHINE ZERO	
5:221	04548 24 05613 06243	AZCK	C	AMPQ, AZ101,,	ZERO MANTISSA CHECK
5:222	04560 47 04596 01200		BNE	ACO	
5:223	04572 16 05619 00099	AZRO	TFM	AEPQ, -99,,	
5:224	04584 49 05148 00000		B	AMKFP,,,	MAKE MACHINE ZERO GOTO MAKE FP
5:225					
5:226					
5:227				* NORMALIZE SHIFT RIGHT	
5:228				* - CARRY OUT- EXP ADJUST	
5:229	04596 16 04710 05397	ACO	TFM	ACMF1+6, AMP,,	INITIALIZE MF INSTRUCTION
5:230	04608 16 04715 05397		TFM	ACMF1+11, AMP	
5:231	04620 16 04758 05397		TFM	ACMF2+6, AMP,,	
5:232	04632 16 04763 05397		TFM	ACMF2+11, AMP	
5:233	04644 43 04668 05621		BD	*+24, AOV	
5:234	04656 49 04824 00000		B	EXADJ,,,	NO CARRY OUT GOTO EXP ADJ
5:235	04668 22 04715 05276		S	*+47, AML	
5:236	04680 11 04715 00001		AM	*+35, 00001	
5:237	04692 22 04710 05276		S	*+18, AML	
5:238	04704 71 05397 05397	ACMF1	MF	AMP, AMP	
5:239	04716 22 04763 05276		S	*+47, AML	
5:240	04728 11 04763 00001		AM	*+35, 00001	

5:241 04740 22 04758 05276 S *+18, AMI
 5:242 04752 71 05505 05505 ACMF2 MF AMQ, AMQ
 5:243 04764 26 05613 06243 TF AMPQ, AZ101,, CLEAR
 5:244 04776 26 05613 05397 TF AMPQ, AMP
 5:245 04788 21 05613 05505 FXAR2 A AMPQ, AMQ,,
 5:246 04800 71 05612 05613 MF AMPQ-1, AMPQ ARGs MAY REPLACE A WITH S
 5:247 04812 26 05613 05612 TF AMPQ, AMPQ-1
 5:248
 5:249
 5:250 * ADJUST EXPONENT
 5:251 * ARITHMETIC OVERFLOW
 5:252 * MACHINE INFINITY
 5:253 * MANTISSA LEFT SHIFT FOR NORMALIZATION
 5:254 04824 26 05619 06042 EXADJ TF AEPQ, AZ5
 5:255 04836 26 05619 05403 TF AEPQ, AEP
 5:256 04848 24 05403 05511 C AEP, AEQ
 5:257 04860 46 04884 01300 BNL *+24
 5:258 04872 26 05619 05511 TF AEPQ, AEQ
 5:259 04884 43 05088 05622 BD AELS, ANRM,, NORMALIZE LEFT SHIFT CHECK
 5:260 04896 43 04920 05621 BD *+24, AOV,,, ARITHMETIC OVERFLOW CHECK
 5:261 04908 49 05148 00000 B AMKFP
 5:262
 5:263 04920 14 05619 00099 CM AEPQ, 00099,, EXP 99 THEN MACH INF
 5:264 04932 46 04968 01300 BNL AINF
 5:265 04944 11 05619 00001 AM AEPQ, 1,, ARITH OV EXP ADJUST
 5:266 04956 49 05148 00000 B AMKFP
 5:267
 5:268 * NORMALIZE SHIFT RIGHT-CARRY OUT
 5:269 * EXPONENT OVERFLOW MACH INFINITY
 5:270 04968 16 05070 05613 AINF TFM AINF2+6, AMPQ,, INITIALIZE SF
 5:271 04980 44 05028 05613 BNF AINF1, AMPQ
 5:272 04992 26 05613 05929 TF AMPQ, A9-3,, MOVE NINES
 5:273 05004 32 05613 00000 SF AMPQ,,, REPLACE SIGN FLAG
 5:274 05016 49 05040 00000 B *+24
 5:275 05028 26 05613 05929 AINF1 TF AMPQ, A9-3,, MOVE NINES
 5:276 05040 22 05070 05276 S *+30, AML
 5:277 05052 11 05070 00001 AM *+18, 00001
 5:278 05064 32 05613 00000 AINF2 SF AMPQ
 5:279 05076 49 05148 00000 B AMKFP
 5:280
 5:281 * NORMALIZING SHIFT LEFT

5:282 * CHECK FOR EXP OVERFLOW
 5:283 * SET TO MACHINE ZERO
 5:284 05088 22 05619 05291 AEELS S AEPQ, AZCNT,, ADJ EXP FOR LEFT SHIFT
 5:285 05100 14 05619 00100 CM AEPQ, -100
 5:286 05112 47 05148 01200 BNE AMKFP
 5:287 05124 16 05619 00099 TFM AEPQ, -99
 5:288 05136 26 05613 06243 TF AMPQ, AZ101
 5:289
 5:290
 5:291 * PUT SUM AND ADJUSTED EXPONENT
 5:292 * INTO FLOATING POINT MM...MM EE
 5:293 05148 71 05618 05615 AMKFP MF AEPQ-1, AEPQ-4
 5:294 05160 26 08153 05619 TF PQFP, AEPQ
 5:295 05172 26 08151 05613 TF PQFP-2, AMPQ
 5:296
 5:297
 5:298 * COMPARE FADD SUM TO A21 SUM
 5:299 * A21RC EQ TRUE IF EQUAL OTW FALSE
 5:300 05184 17 09922 05212 BTM CF, *+28,, COMPARE FLOATING SUBROUTINE
 5:301 05200 063600..306366 DSA A21D1, PQFP, A21RC
 5:302
 5:303
 5:304 * SEND RETURN CODE TO CALLING PROGRAM
 5:305 05212 26 05247 03215 TF *+35, A21-1,,
 5:306 05224 12 05247 00002 SM *+23, 2,,
 5:307 05236 26 05254 00000 TF *+18, 0,,
 5:308 05248 25 00000 06366 TD 0, A21RC,,
 5:309
 5:310
 5:311 * RETURN
 5:312 05260 49 03215 00000 FACR B A21-1,,,6, RETURN TO CALLING PROGRAM
 5:313
 5:314
 5:315
 5:316
 5:317
 5:318 * ----- WORK AREAS FOR A (21) INSTRUCTION
 5:319 05276 00000 AML DC 5, 0,,, MANTISSA LENGTH

5:320	05281	$\bar{0}0000$	AED	DC	5, 0,,,	EXP DIFFERENCE D = ABS (EP-EQ)
5:321						
5:322	05286	$\bar{0}0000$	AFLG	DC	5, 0,,,	MANTISSA FLAG ADDRESS
5:323	05291	$\bar{0}0000$	AZCNT	DC	5, 0,,,	MANTISSA LEADING ZEROS COUNT
5:324						
5:325	05296	$\bar{0}0000$	AMPA	DC	5, 0,,,	POINTER TO AMP
5:326						
5:327	05346	$\bar{0}00000..000000$		DC	50, 0,,,	MP MANTISSA FIELD
5:328	05347	$000000..000000$		DSC	50, 0,,,	SIZE IS 101 FOR CARRY/OVERFLOW
5:329	05397	0	AMP	DSC	1, 0	
5:330	05398	@		DC	1, @	
5:331						
5:332	05403	$\bar{0}0000$	AEP	DC	5, 0,,,	P EXPONENT
5:333	05404	@		DC	1, @	
5:334						
5:335	05454	$\bar{0}00000..000000$		DC	50, 0,,,	MQ MANTISSA FIELD
5:336	05455	$000000..000000$		DSC	50, 0,,,	SIZE IS 101 FOR CARRY/OVERFLOW
5:337	05505	0	AMQ	DSC	1, 0	
5:338	05506	@		DC	1, @	
5:339						
5:340	05511	$\bar{0}0000$	AEQ	DC	5, 0,,,	Q EXPONENT
5:341	05512	@		DC	1, @	
5:342						
5:343	05562	$\bar{0}00000..000000$		DC	50, 0,,,	MP + MQ SUM MANTISSA FIELD
5:344	05563	$000000..000000$		DSC	50, 0,,,	
5:345	05613	0	AMPQ	DSC	1, 0	
5:346	05614	@		DC	1, @	
5:347						
5:348	05619	$\bar{0}0000$	AEPQ	DC	5, 0,,,	FINAL EXPONENT
5:349	05620	@		DC	1, @	
5:350						
5:351	05621	$\bar{0}$	AOV	DC	1, 0,,,	OVERFLOW FLAG FOR CARRY OUT
5:352	05622	$\bar{0}$	ANRM	DC	1, 0,,,	NORMALIZE FLAG

5:386	06248	0 0000	AMPA A DC	5, 0,,,	HIGH ORDER ADDRESS OF AMPMA
5:387					
5:388	06253	0 0000	A21A1 DC	5, 0,,,	ARG1
5:389	06258	0 0000	A21A2 DC	5, 0,,,	ARG2
5:390					
5:391	06308	0 00000..000000	DC	50, 0	
5:392	06309	000000..000000	DSC	50, 0	
5:393	06360	0 0	A21D1 DC	2, 0 ,,,	FADD CALLING PROGRAM PRODUCT
5:394	06361	@	DC	1, @	
5:395					
5:396	06366	0 0000	A21RC DC	5, 0,,,	RETURN CODE
5:397					

----- 6: fp40-fmul.sps -----

6:1	* =====				
6:2	* FMUL TEST				
6:3	*				
6:4	*FMUL				
6:5	* =====				
6:6					
6:7	06371	00005	DS	5,,,	RETURN ADDRESS FROM BTM
6:8	06372	41 00000 00000	FMTST NOP	00000, 00000	
6:9					
6:10	06384	34 00000 00102	RCTY		
6:11	06396	34 00000 00102	RCTY		
6:12	06408	39 13551 00100	WATY	FMULT,,,	MAIN FMUL TEST TITLE
6:13	06420	34 00000 00102	RCTY		
6:14					
6:15					
6:16	* INITIALIZE TABLE POINTER				
6:17					
6:18	06432	16 13644	0 0000	TFM	FMINC, 0,, TABLE POINTER
6:19					
6:20	* POINT TO TEST DATA TABLES				
6:21	* RESET				
6:22	06444	16 13664	1 3694	TFM	FMTBL, FMTBI

6:23	06456 16 13669	<u>1</u> 3734	TFM	FMTBL+5, FMTBP	
6:24	06468 16 13674	<u>1</u> 3774	TFM	FMTBL+10, FMTBQ	
6:25	06480 26 13659	13664	TF	FMTP, FMTBL	
6:26	06492 26 13664	13669	TF	FMTP+5, FMTBL+5	
6:27	06504 26 13669	13674	TF	FMTP+10, FMTBL+10	
6:28					
6:29	06516 41 00000	00000	FMNXT	NOP	
6:30					
6:31	06528 21 13659	13644	A	FMTP, FMINC,,	ID MESSAGE TABLE
6:32	06540 14 1365 <u>9</u>	00000	CM	FMTP, 00000, 6,	RETURN ON ZERO ADDRESS
6:33	06552 46 0637 <u>1</u>	01200	BE	FMTST-1,,6,	RETURN
6:34					
6:35	06564 21 13664	13644	A	FMTP+5, FMINC,,	MULTIPLICAND TABLE
6:36	06576 21 13669	13644	A	FMTP+10, FMINC,,	MULTIPLIERS TABLE
6:37					
6:38	06588 16 13644	<u>0</u> 0005	TFM	FMINC, 5	
6:39					
6:40	06600 26 13679	<u>1</u> 3659	TF	FMI, FMTP,11	
6:41	06612 26 13684	<u>1</u> 3664	TF	FMP, FMTP+5,11	
6:42	06624 26 13689	<u>1</u> 3669	TF	FMQ, FMTP+10,11	
6:43					
6:44				* START OF CURRENT FMUL TEST	
6:45	06636 41 00000	00000	FMRPT	NOP	
6:46	06648 34 00000	00102		RCTY	
6:47	06660 39 1367 <u>9</u>	00100	WATY	FMI,,6,	TEST TITLE
6:48	06672 34 00000	00101		SPTY	
6:49					
6:50				*--- NEED TO PRESERVE MULTIPLICAND - MOVE TO FMPRD FIRST	
6:51				*--- AVOID TFL FOR FLOATING POINT TESTS	
6:52					
6:53	06684 26 13911	<u>1</u> 3684	TF	FMPRD, FMP, 11, FMP POINTS TO EXPONENT	
6:54	06696 12 13684	<u>0</u> 0002	SM	FMP, 00002,,	FMP NOW POINTS TO MANTISSA
6:55	06708 26 13909	<u>1</u> 3684	TF	FMPRD-2, FMP, 11,	

6:56 06720 11 13684 00002 AM FMP, 00002,, FMA AGAIN POINTS TO EXPONENT
 6:57

6:58 06732 03 13911 13689 FMUL FMPRD, FMQ, 11

6:59

6:60 * INDICATOR CHECK

6:61 06744 47 06768 01400 BNV *+24

6:62 06756 39 15015 00100 WATY OVM1,,, OVERFLOW MESSAGE

6:63 06768 47 06792 01500 BNXV *+24

6:64 06780 39 15037 00100 WATY XCKM1,,, EXPONENT OVERFLOW MESSAGE

6:65

6:66 * CALL SUBROUTINE M23

6:67 * M (23) MANTASSAS AND COMPUTE EXPONENTS

6:68 06792 17 06900 06824 BTM M23, *+32,, TRANSMIT RETURN ADDRESS

6:69 06808 136841..14955 DSA -FMP, -FMQ, FMPRD, STAT

6:70

6:71

6:72 * CHECK COMPARE STATUS

6:73 * CALL ERROR ROUTINE AND RETURN ITS STATUS IN ESTAT

6:74 06824 17 09518 06846 BTM ERTN, *+22,, TRANSMIT ADDRESS OF NEXT INSTRUCTION

6:75 06840 1495514954 DSA STAT, ESTAT

6:76

6:77 06846 43 06870 14954 BD FMEND, ESTAT

6:78 06858 49 06636 00000 B FMRPT,,, RERUN TEST

6:79

6:80 06870 41 00000 00000 FMEND NOP 00000, 00000

6:81 06882 49 06516 00000 B FMNXT,,, GOTO NEXT FMUL TEST

6:82

----- 7: fp41-fixmul.sps -----

7:1 *

7:2 * -----

7:3 * SUBROUTINE STATUS = M23(FP1, FP2, FP-PRODUCT)

7:4 * COMPARES FMUL PRODUCT WITH M(23)/EX-SUM

7:5 * REF (1) 227-5630-1 IBM 1620 FLOATING POINT FEATURE CE MANUAL

7:6 *

7:7 * VERIFIES FMUL RESULT BY USING M(23) TO MULTIPLY MANTISSAS

7:8 * AND SEPARATELY ADDS EXPONENTS.

7:9 * FLOATING POINT NUMBER IS CREATED FROM M(23) PRODUCT AND

* (ADJUSTED) SUM OF EXPONENTS

```

7:10
7:11          *
7:12          * HANDLES CASES DESCRIBED IN REF (1) FLOATING MULTIPLY EXAMPLE
7:13          *
7:14          * FMUL EXAMPLE 1. NO EXPONENT MODIFY REQUIRED
7:15          * FMUL EXAMPLE 2. EXPONENT MODIFY REQUIRED
7:16          * FMUL EXAMPLE 3. DIGIT FORCE -- MACHINE INFINITY
7:17          * FMUL EXAMPLE 4. EXPONENT OVERFLOW -- MODIFY PE TO 99
7:18          * FMUL EXAMPLE 5. SPECIAL CASE 999 X 199
7:19          *
7:20          *
7:21          * USAGE
7:22          *      BTM    M23,   *+32,,           TRANSMIT RETURN ADDRESS
7:23          *      DSA    -FMP,  -FMQ, FMPRD, STAT
7:24          *      NOP    00000, 00000,,       RETURN TO HERE
7:25          *
7:26          * M23A1 = -FMP (INDIRECT ADDRESS) FLOATING POINT NUMBER
7:27          * M23A2 = -FMQ (INDIRECT ADDRESS) FLOATING POINT NUMBER
7:28          * M23D1 = FMPRD (DATA FIELD) FLOATING POINT FMUL PRODUCT
7:29          * M23RC = STAT ( DATA, RETURN CODE )
7:30          * RETURNS STATUS OF COMPARE
7:31          *
7:32          *
7:33          *M23
7:34          * -----
7:34      06898 00005          DS    5,,           RETURN ADDRESS
7:35      06900 41 00000 00000  M23    NOP    00000, 00000
7:36
7:37          * GET FIRST ARGUMENT (INDIRECT)
7:38      06912 26 06947 06899  TF    *+35, M23-1,,

7:39      06924 12 06947 00016    SM    *+23, 00016,,,
7:40      06936 26 06971 00000    TF    *+35, 0,,,
7:41      06948 33 06971 00000    CF    *+23
7:42      06960 26 08498 00000    TF    M23A1,  0,,,
7:43
7:44
7:45          * GET SECOND ARGUMENT (INDIRECT)
7:46      06972 26 07007 06899  TF    *+35, M23-1,,,

7:47      06984 12 07007 00011    SM    *+23, 00011,,,
7:48      06996 26 07031 00000    TF    *+35, 0,,,
7:49      07008 33 07031 00000    CF    *+23
7:50      07020 26 08503 00000    TF    M23A2,  0,,,
7:51
7:52

```

7:53 * GET THIRD ARGUMENT
 7:54 07032 26 07067 06899 TF *+35, M23-1,,
 7:55 07044 12 07067 00006 SM *+23, 00006,,
 7:56 07056 26 07079 00000 TF *+23, 0,,
 7:57 07068 26 08605 00000 TF M23D1, 0,, -- GETS EXP
 7:58
 7:59 07080 26 07115 07079 TF *+35, *-1,,
 7:60 07092 12 07115 00002 SM *+23, 00002,
 7:61 07104 26 08603 00000 TF M23D1-2, 0,, -- GETS MANTISSA
 7:62
 7:63
 7:64 * ADD MP AND MQ EXPONENTS
 7:65 07116 26 08159 08373 TF SUMEX, MZ5,, CLEAR
 7:66 07128 21 08159 08498 A SUMEX, M23A1, 11, EXPONENT 1
 7:67 07140 21 08159 08503 A SUMEX, M23A2, 11, EXPONENT 2
 7:68
 7:69 * MANTISSA
 7:70 07152 12 08498 00002 SM M23A1, 00002
 7:71 07164 12 08503 00002 SM M23A2, 00002
 7:72 07176 26 07848 08498 TF MPMA, M23A1, 11, MANTISSA 1
 7:73 07188 26 07949 08503 TF MQMA, M23A2, 11, MANTISSA 2
 7:74
 7:75 * MANTISSA LENGTH OF FIRST ARG
 7:76 * L REPRESENTS MANTISSA LENGTH
 7:77 07200 16 07748 00001 TFM FML, 00001
 7:78 07212 26 08493 08498 TF MPMAA, M23A1
 7:79
 7:80 07224 12 08493 00001 ML SM MPMAA, 1
 7:81 07236 11 07748 00001 AM FML, 1
 7:82 07248 44 07224 08493 BNF ML, MPMAA, 11
 7:83
 7:84 * IF EXP SUM GT 100 THEN RESULT IS MACHINE INFINITY 99...9 99
 7:85 * IF EXP SUM EQ 100 ONLY SET EXP TO 99

7:86 07260 14 08159 $\bar{0}0100$ CM SUMEX, 100
 7:87 07272 47 07392 01300 BL MUL,,, SUM EXP LT 100
 7:88 07284 46 07380 01200 BE ME100,,, SUM EXP EQ 100
 7:89
 7:90 07296 26 08159 08262 TF SUMEX, M9,, SUM EXP GT 100 SET SUM EXP
 7:91 07308 26 08363 08260 TF MWK1-2, M9-2,, COPY 9S TO WORK AREA
 7:92 07320 22 07350 07748 S *+30, FML
 7:93 07332 11 07350 $\bar{0}0001$ AM *+18, 00001,
 7:94 07344 32 08363 00000 SF MWK1-2
 7:95 07356 26 08050 08363 TF PQMA, MWK1-2,, 9S IN MANTISSA PRODUCT F
 7:96 07368 49 07596 00000 B MKFP,,, GOTO MAKE FP NUM FOR COM
 7:97
 7:98 07380 26 08159 08262 ME100 TF SUMEX, M9,, SUM EXP EQ 100 SET SUM EXP
 7:99
 7:100 * MULTIPLY MANTISSAS (P,Q) USING M (23)
 7:101 07392 26 08478 $\bar{0}8424$ MUL TF PAREA, MZ100,6, 100 ZEROS TO PRODUCT AR
 7:102 07404 23 07848 07949 M MPMA, MQMA,, P MANTISSA X Q MANTISSA
 7:103
 7:104 * CHECK FOR EXPONENT CASE - PRODUCT LENGTH LT 2L
 7:105 * HIGH ORDER 0
 7:106 * PROD-HIGH = 100 -2L
 7:107 07416 16 08483 $\bar{0}0100$ TFM PAR1, 00100,, PRODUCT AREA (99) + 1
 7:108 07428 22 08483 07748 S PAR1, FML,, 100 - L
 7:109 07440 22 08483 07748 S PAR1, FML,, 100 - L
 7:110 07452 43 07548 08483 BD PRSH, PAR1,11, IF HIGH ORDER 0 THEN MOV
 7:111 07464 26 08488 08483 TF PARW, PAR1
 7:112 07476 11 08488 $\bar{0}0001$ AM PARW, 00001,
 7:113 07488 71 08488 $\bar{0}8483$ MF PARW, PAR1,611
 7:114 07500 12 07748 $\bar{0}0001$ SM FML, 00001,
 7:115
 7:116 07512 14 08159 $\bar{0}0099$ CM SUMEX, 00099,, IF OV KEEP 99
 7:117 07524 46 07548 01200 BE PRSH
 7:118 07536 12 08159 $\bar{0}0001$ SM SUMEX, 00001,, MODIFY EXP IF HI ORD DIGIT
 7:119
 7:120 * SHIFT PRODUCT L (MANTISSA LENGTH) AND MOVE SIGN FLAG

7:121 07548 26 08488 08478 PRSH TF PARW, PAREA
 7:122 07560 22 08488 07748 S PARW, FML,,
 7:123 07572 71 08488 08478 MF PARW, PAREA, 611
 7:124
 7:125 * MOVE PRODUCT TO RESULT AREA
 7:126 07584 26 08050 08488 TF PQMA, PARW , 11,
 7:127
 7:128 * PUT PRODUCT AND EXPONENT SUM INTO FLOATING POINT MM...MM F
 7:129 07596 41 00000 00000 MKFP NOP 0000,0000
 7:130 07608 71 08158 08155 MF SUMEX-1, SUMEX-4
 7:131 07620 26 08153 08159 TF PQFP, SUMEX
 7:132 07632 26 08151 08050 TF PQFP-2, PQMA
 7:133
 7:134 * COMPARE FMUL PRODUCT TO M23 PRODUCT
 7:135 * M23RC EQ TRUE IF EQUAL OTW FALSE
 7:136 07644 17 09922 07672 BTM CF, *+28,, TRANSMIT RETURN ADDRESS
 7:137 07660 086050..308607 DSA M23D1, PQFP, M23RC
 7:138
 7:139 * SEND RETURN CODE TO CALLING PROGRAM
 7:140 07672 26 07707 06899 TF *+35, M23-1,,
 7:141 07684 12 07707 00001 SM *+23, 1,,
 7:142 07696 26 07714 00000 TF *+18, 0,,
 7:143 07708 25 00000 08607 TD 0, M23RC,,
 7:144
 7:145 07720 41 00000 00000 FMCR NOP 00000, 00000
 7:146
 7:147 * RETURN
 7:148 07732 49 06899 00000 B M23-1,, 6, RETURN TO CALLING PROGRAM
 7:149
 7:150 * ----- WORK AREAS FOR M (23) INSTRUCTION
 7:151 07748 00000 FML DC 5, 0,,, MANTISSA LENGTH
 7:152
 7:153 07798 000000..000000 DC 50, 0,,, MP MANTISSA FIELD
 7:154 07799 000000..000000 DSC 49, 0,,,
 7:155 07848 0 MPMA DSC 1, 0,,,
 7:156 07849 @ DC 1, @
 7:157

7:191
7:192

7:193 08478 ~~00099~~ PAREA DC 5, 00099,, MACHINE PRODUCT AREA

7:194 08483 ~~00100~~ PAR1 DC 5, 00100,,

7:195 08488 ~~00099~~ PARW DC 5, 00099,, MACHINE PRODUCT AREA

7:196

7:197 08493 ~~00000~~ MPMAA DC 5, 0,,, HIGH ORDER ADDRESS OF MPMA

7:198

7:199 08498 ~~00000~~ M23A1 DC 5, 0,,, ARG1

7:200 08503 ~~00000~~ M23A2 DC 5, 0,,, ARG2

7:201

7:202 08553 ~~000000..000000~~ DC 50, 0

7:203 08554 000000..000000 DSC 50, 0

7:204 08605 ~~00~~ M23D1 DC 2, 0,,, FMUL CALLING PROGRAM PRODUCT

7:205 08606 @ DC 1, @

7:206

7:207 08607 ~~0~~ M23RC DC 1, 0,,, RETURN CODE

----- 8: fp50-subroutines.sps -----

8:1 * =====

8:2 * SET SENSE SWITCHES SUBROUTINE

8:3 * =====

8:4

8:5 08612 00005 DS 5,,, RETURN ADDRESS FROM BTM

8:6 08614 41 00000 00000 SETS1 NOP

8:7 * CALL SUBROUTINE PRINT SENSE SWITCH STATUS

8:8 08626 17 08704 ~~08638~~ BTM SWSTA, *+12,, TRANSMIT RETURN ADDRESS

8:9 08638 34 00000 00102 RCTY

8:10

8:11 * SET SENSE SWITCHES AND CONTINUE

8:12 08650 39 09341 00100 WATY SWMS5

8:13 08662 34 00000 00102 RCTY

8:14 08674 48 00000 00000 H

8:15

8:16 08686 49 08613 00000 B SETS1-1,,6, RETURN
8:17
8:18
8:19 * -----
8:20 * SUBROUTINE PRINT SENSE SWITCH STATUS
8:21 *
8:22 *SWSTA
8:23 * -----
8:24
8:25 08702 00005 DS 5,, SWATA RETURN ADDRESS
8:26 08704 41 00000 00000 SWSTA NOP
8:27 08716 34 00000 00102 RCTY
8:28 08728 39 09185 00100 WATY SWMS2
8:29 08740 34 00000 00102 RCTY
8:30
8:31 08752 39 09245 00100 WATY SWMS3
8:32 08764 34 00000 00102 RCTY
8:33
8:34 08776 39 09291 00100 WATY SWMS4
8:35 08788 34 00000 00102 RCTY
8:36
8:37 08800 39 09065 00100 WATY SWST,,, SENSE SWITCH STATUS
8:38 08812 34 00000 00102 RCTY
8:39 08824 39 09105 00100 WATY SWMS1
8:40 08836 34 00000 00102 RCTY
8:41
8:42 08848 46 08884 00100 BC1 *+36,,, SW1
8:43 08860 39 09169 00100 WATY OFF
8:44 08872 49 08896 00000 B *+24
8:45 08884 39 09153 00100 WATY ON
8:46
8:47 08896 46 08932 00200 BC2 *+36,,, SW2
8:48 08908 39 09169 00100 WATY OFF
8:49 08920 49 08944 00000 B *+24
8:50 08932 39 09153 00100 WATY ON
8:51
8:52 08944 46 08980 00300 BC3 *+36,,, SW3
8:53 08956 39 09169 00100 WATY OFF
8:54 08968 49 08992 00000 B *+24
8:55 08980 39 09153 00100 WATY ON
8:56
8:57 08992 46 09028 00400 BC4 *+36,,, SW4
8:58 09004 39 09169 00100 WATY OFF
8:59 09016 49 09040 00000 B *+24

8:60 09028 39 09153 00100 WATY ON
8:61
8:62 09040 34 00000 00102 RCTY
8:63
8:64 09052 49 08703 00000 B SWSTA-1,,6, RETURN TO CALLING PROGRAM
8:65
8:66
8:67 09065 624555..64620@ SWST DAC 20, SENSE SWITCH STATUS@
8:68
8:69 09105 007100..00740@ SWMS1 DAC 24, 1 2 3 4@
8:70 09153 565500..00000@ ON DAC 8,ON @
8:71 09169 564646..00000@ OFF DAC 8,OFF @
8:72
8:73 09185 006266..56590@ SWMS2 DAC 30, SW1 ON = DO NOT PRINT ERROR@
8:74 09245 006266..62630@ SWMS3 DAC 23, SW2 ON = REPEAT TEST@
8:75 09291 006266..56590@ SWMS4 DAC 25, SW3 ON = HALT ON ERROR@
8:76 09341 624563..00000@ SWMS5 DAC 43, SET SENSE SWITCHES THEN CONTINUE@
8:77 09427 594559..00000@ SWMS6 DAC 43,RERUN MODE. SW2 OFF = EXIT RERUN@
8:78
8:79
8:80
8:81
8:82 * -----
8:83 * SUBROUTINE FL COMMAND TESTS ERROR ROUTINE
8:84 * ERRC = ERTN(FL-CMD-TEST-STAT)
8:85 *
8:86 * CALLING
8:87 * BTM ERTN, *+23,, TRANSMIT ADDRESS OF NEXT INSTRUCTION
8:88 * DSA FL-CMD-TEST-STAT, ESTAT
8:89 * NEXT INSTRUCTION
8:90 *
8:91 * ARGS
8:92 * FL-CMD-TEST-STAT IS EA1
8:93 *
8:94 * RETURN CODE

8:95 * ERRC 0 = RERUN TEST 1 = GOTO NEXT TEST
 8:96 *
 8:97 * ERROR ROUTINE
 8:98 * SW1 ON = DO NOT PRINT ERROR
 8:99 * SW2 ON = REPEAT TEST
 8:100 * SW3 ON = HALT ON ERROR
 8:101 *
 8:102 *ERTN
 8:103 * -----
 8:104
 8:105 09516 00005 DS 5,, ERTN RETURN ADDRESS
 8:106 09518 41 00000 00000 ERTN NOP
 8:107
 8:108
 8:109 * GET FIRST ARGUMENT
 8:110 09530 26 09565 09517 TF *+35, ERTN-1,,
 8:111 09542 12 09565 00006 SM *+23, 6,,
 8:112 09554 26 09577 00000 TF *+23, 0,,
 8:113 09566 26 09914 00000 TF EA1, 0,, FL-CMD-TEST-STAT 1=PASS 0=FAIL
 8:114
 8:115 09578 25 09915 14953 TD ERRC, FALSE,, RET CD DEFAULT IS RERUN TEST
 8:116
 8:117 09590 43 09662 09914 BD ERPAS, EA1,, CHECK PASS OR FAIL
 8:118
 8:119 09602 46 09626 00100 BC1 ERSW3
 8:120 09614 39 14957 00100 WATY FAIL
 8:121
 8:122 09626 47 09686 00300 ERSW3 BNC3 ERSW2
 8:123 09638 48 00000 00000 H 00000, 00000
 8:124 09650 49 09686 00000 B ERSW2
 8:125
 8:126 09662 41 00000 00000 ERPAS NOP 00000, 00000
 8:127 09674 39 14991 00100 WATY PASS
 8:128
 8:129 09686 41 00000 00000 ERSW2 NOP
 8:130 09698 46 09734 00200 BC2 ERPSW,,, IF SW2 ON REPEAT TEST
 8:131 09710 25 09915 14952 TD ERRC, TRUE,, GOTO NEXT TEST
 8:132 09722 49 09842 00000 B ERRT,,, RETURN
 8:133
 8:134 * CALL SUBROUTINE PRINT SENSE SWITCH STATUS
 8:135 09734 17 08704 09746 ERPSW BTM SWSTA, *+12,, TRANSMIT RETURN ADDRESS
 8:136 09746 34 00000 00102 RCTY
 8:137

```

8:138          * SET SENSE SWITCHES AND CONTINUE
8:139    09758 34 00000 00102      RCTY
8:140    09770 34 00000 00102      RCTY
8:141    09782 39 09427 00100      WATY   SWMS6
8:142    09794 34 00000 00102      RCTY
8:143    09806 39 09341 00100      WATY   SWMS5
8:144    09818 34 00000 00102      RCTY
8:145    09830 48 00000 00000      H

8:146
8:147
8:148    09842 41 00000 00000  ERRT  NOP
8:149                      * MOVE RETURN CODE TO THIRD ARGUMENT
8:150    09854 26 09889 09517      TF    *+35, ERTN-1,,

8:151    09866 12 09889 00001      SM    *+23, 1,,,
8:152    09878 26 09896 00000      TF    *+18, 0,,,
8:153    09890 25 00000 09915      TD    0, ERRC,,

8:154
8:155                      * RETURN

8:156    09902 49 09517 00000      B     ERTN-1,,6,           RETURN TO CALLING PROG
8:157
8:158                      * =====

8:159    09914 0             EA1   DC    1, 0,,,       FL-CMD-TEST-STAT
8:160    09915 0             ERRC  DC    1, 0,,,       ERROR RTN RETURN CODE 0 = RERUN TEST

8:161
8:162
8:163                      * -----
8:164                      * SUBROUTINE STATUS = CF(FP1, FP2)
8:165                      *                   COMPARE FLOATING
8:166
8:167                      * COMPARES TWO FLOATING POINT NUMBERS BY
8:168                      *       COMPARING THE MANTISSA FIELDS AND EXPONENT FIELDS
8:169
8:170                      * CALLING
8:171                      *       BTM   CF, *+28,, TRANSMIT ADDRESS OF NEXT INSTRUCTION
8:172                      *       DSA   FP1, FP2, STATUS
8:173                      *       NEXT INSTRUCTION
8:174
8:175                      * CFA1   ADDRESS OF FLOATING POINT NUMBER 1
8:176
8:177                      * CFA2   ADDRESS OF FLOATING POINT NUMBER 2
8:178

```

```

8:179          * CFRC  ADDRESS OF RETURN CODE (STATUS)
8:180          *           TRUE = EQ, FALSE = NE
8:181          *
8:182          * RETURN ADDRESS (BTM)  CF-1
8:183          *
8:184          *CF
8:185          * -----
8:186
8:187 09920 00005      DS   5,,          CF RETURN ADDRESS
8:188 09922 41 00000 00000  CF   NOP   00000, 00000
8:189
8:190          * GET FIRST ARGUMENT
8:191 09934 26 09969 09921  TF   *+35, CF-1,,,
8:192 09946 12 09969 00012  SM   *+23, 12,,,
8:193 09958 26 09981 00000  TF   *+23, 0,,,
8:194 09970 16 10310 00000  TFM  CFA1, 0,,,
8:195 09982 44 10030 09981  BNF  CFSEC, *-1,,,
8:196 09994 26 10029 09981  TF   *+35, *-13
8:197 10006 33 10029 00000  CF   *+23
8:198 10018 26 10310 00000  TF   CFA1, 0
8:199
8:200          * GET SECOND ARGUMENT
8:201 10030 26 10065 09921  CFSEC TF   *+35, CF-1,,,
8:202 10042 12 10065 00007  SM   *+23, 7,,,
8:203 10054 26 10077 00000  TF   *+23, 0,,,
8:204 10066 16 10315 00000  TFM  CFA2, 0,,,
8:205 10078 44 10126 10077  BNF  CFINI, *-1,,,
8:206 10090 26 10125 10077  TF   *+35, *-13
8:207 10102 33 10125 00000  CF   *+23
8:208 10114 26 10315 00000  TF   CFA2, 0
8:209
8:210
8:211          * INITIALIZATION
8:212 10126 25 10316 14952  CFINI TD   CFRC, TRUE,,,
8:213
8:214          * COMPARE EXPONENTS
8:215 10138 24 10310 10315  C    CFA1, CFA2, 611,
8:216 10150 46 10174 01200  BE   MANT,,,,
8:217 10162 25 10316 14953  TD   CFRC, FALSE,,,
8:218

```

8:219 * GET MANTISSA ADDRESSES

8:220 10174 12 10310 00002 MANT SM CFA1, 00002,, BACKUP 2 TO GET MANTISSA

8:221 10186 12 10315 00002 SM CFA2, 00002,, BACKUP 2 TO GET MANTISSA

8:222

8:223 * COMPARE MANTASSAS

8:224 10198 24 10310 10315 C CFA1, CFA2, 611

8:225 10210 46 10234 01200 BE CFRT,,, RETURN IF EQ

8:226 10222 25 10316 14953 TD CFRC, FALSE,, SET RETURN CODE

8:227

8:228 10234 41 00000 00000 CFRT NOP

8:229 * MOVE RETURN CODE (CFRC) TO THIRD ARGUMENT

8:230 10246 26 10281 09921 TF *+35, CF-1,,

8:231 10258 12 10281 00002 SM *+23, 2,,

8:232 10270 26 10288 00000 TF *+18, 0,,

8:233 10282 25 00000 10316 TD 0, CFRC,,

8:234

8:235 * RETURN

8:236 10294 49 09921 00000 B CF-1,,6, RETURN TO CALLING PROGRAM

8:237

8:238 * =====

8:239 10310 00000 CFA1 DC 5, 0

8:240 10315 00000 CFA2 DC 5, 0

8:241 10316 0 CFRC DC 1, 0,,, RETURN CODE 1=TRUE 0=FALSE

8:242

8:243

----- 9: fp60-symbols-dend.sps -----

9:1 * =====

9:2 * =====

9:3 * -----

9:4 * SYMBOLS

9:5 * -----

9:6 * =====

9:7

9:8 10319 465356..59630@ TITLE DAC 35,FLOATING POINT DIAGNOSTICS - START@

9:9 10389 465356..55440@ DONE DAC 33, FLOATING POINT DIAGNOSTICS - END@
9:10
9:11
9:12 * WORK AREAS
9:13

9:14 10503 000000..000000 DC 50, 0
9:15 10504 000000..000000 DSC 50, 0
9:16 10554 00 WK1Z DSC 2, 0,,, TO ZERO WK1
9:17 10556 @ DC 1,@
9:18
9:19 10557 000000..000000 DSC 50, 0
9:20 10607 000000..000000 DSC 50, 0

9:21 10658 00 WK1 DC 2, 0
9:22 10659 @ DC 1,@
9:23
9:24 10660 000000..000000 DSC 50, 0
9:25 10710 000000..000000 DSC 50, 0

9:26 10761 00 WK2 DC 2, 0
9:27 10762 @ DC 1,@
9:28
9:29 *TFL, BTFL, FSR, FSL DATA START
9:30
9:31

9:32 10767 00000 TFINC DC 5, 0,,, TABLE POINTER
9:33
9:34 * TABLE POINTER
9:35 10777 00010 TFTBP DS 10
9:36
9:37 * LIST OF DATA TABLES

9:38 10782 1080210837 TFTBL DSA TFTBI, TFTBD
9:39
9:40
9:41 10792 00005 TFI DS 5
9:42 10797 00005 TFD DS 5
9:43
9:44 * TF TEST IDENTIFICATION MESSAGES
9:45
9:46 10802 108691..100000 TFTBI DSA TFI1, TFI2, TFI3, TFI4, TFI5,

9:117 11748 119431..812053 FSTBL DSA FSTBI, FSTBP, FSTBQ
9:118
9:119
9:120 11763 00005 FAI DS 5
9:121 11768 00005 FAP DS 5
9:122 11773 00005 FAQ DS 5
9:123
9:124 * FADD TEST IDENTIFICATION MESSAGES
9:125 11778 122091..713439 FATBI DSA FAI1, FAI2, FAI3, FAI4, FAI5, FAI7, FAI8, FAI9, FAI10
9:126 11828 00000 DSA 00000
9:127
9:128 * P DATA
9:129 11833 123121..013542 FATBP DSA FAP1, FAP2, FAP3, FAP4, FAP5, FAP7, FAP8, FAP9, FAP10
9:130 11883 00000 DSA 00000
9:131
9:132 * Q DATA
9:133 11888 123181..613548 FATBQ DSA FAQ1, FAQ2, FAQ3, FAQ4, FAQ5, FAQ7, FAQ8, FAQ9, FAQ10
9:134 11938 00000 DSA 00000
9:135
9:136
9:137 ***** FSUB TEST DATA 7 AND 8 DIFFER -- P NEGATIVE
9:138 * FSUB TEST IDENTIFICATION MESSAGES
9:139 11943 122091..713439 FSTBI DSA FAI1, FAI2, FAI3, FAI4, FAI5, FSI7, FSI8, FAI9, FAI10
9:140 11993 00000 DSA 00000
9:141
9:142 * P DATA
9:143 11998 123121..013542 FSTBP DSA FAP1, FAP2, FAP3, FAP4, FAP5, FSP7, FSP8, FAP9, FAP10
9:144 12048 00000 DSA 00000
9:145
9:146 * Q DATA
9:147 12053 123181..613548 FSTBQ DSA FAQ1, FAQ2, FAQ3, FAQ4, FAQ5, FSQ7, FSQ8, FAQ9, FAQ10
9:148 12103 00000 DSA 00000

9:149
 9:150
 9:151
 9:152 * FADD DATA SYMBOLS
 9:153

9:154	12153	$\bar{0}00000..000000$	DC	50, 0
9:155	12154	000000..000000	DSC	50, 0
9:156	12205	$\bar{0}0$	FASUM DC	2, 0
9:157	12206	@	DC	1, @
9:158			* REF 1. 227-5630-1 IBM 1620 FLOATING POINT FEATURE CE MANUAL	
9:159			* EXAMPLES 1-5 EXPONENT CONFIGURATION	
9:160			* EXAMPLES 6-11 FRACTION (MANTISSA) CONFIGURATION	
9:161			* EXAMPLES 1 AND 6 SAME	
9:162				
9:163				
9:164				
9:165	12209	$\bar{4}56771..64630@$	FAI1 DAC	50, EX1 PE = QE / D = 0 NO NORMALIZING - NO CARRY
9:166				
9:167	12310	$\bar{1}23$	DC	3, 123
9:168	12312	$\bar{0}4$	FAP1 DC	2, 04
9:169	12313	@	DC	1, @
9:170				
9:171	12316	$\bar{2}46$	DC	3, 246
9:172	12318	$\bar{0}4$	FAQ1 DC	2, 04
9:173	12319	@	DC	1, @
9:174				
9:175				
9:176	12321	$\bar{4}56772..43490@$	FAI2 DAC	50, EX2 PE GT QE - D LT L QF SHIFTED RT TO ALIGN I
9:177				
9:178	12422	$\bar{1}23$	DC	3, 123
9:179	12424	$\bar{0}6$	FAP2 DC	2, 06
9:180	12425	@	DC	1, @
9:181				
9:182	12428	$\bar{2}46$	DC	3, 246

9:183 12430 04 FAQ2 DC 2, 04
9:184 12431 @ DC 1, @
9:185
9:186
9:187 12433 456773..00000@ FAI3 DAC 50,EX3 PE GT QE - D GT L - Q DISCARDED-RESULT IS
9:188
9:189 12534 123 DC 3, 123
9:190 12536 08 FAP3 DC 2, 08
9:191 12537 @ DC 1, @
9:192
9:193 12540 246 DC 3, 246
9:194 12542 03 FAQ3 DC 2, 03
9:195 12543 @ DC 1, @
9:196
9:197
9:198 12545 456774..49540@ FAI4 DAC 50,EX4 PE LT QE- D LT L PF SHIFTED RT TO ALIGN DE
9:199
9:200 12646 123 DC 3, 123
9:201 12648 02 FAP4 DC 2, 02
9:202 12649 @ DC 1, @
9:203
9:204 12652 246 DC 3, 246
9:205 12654 04 FAQ4 DC 2, 04
9:206 12655 @ DC 1, @
9:207
9:208
9:209 12657 456775..00000@ FAI5 DAC 50,EX5 PE LT QE - D GT L P DISCARDED-RESULT IS Q
9:210
9:211 12758 123 DC 3, 123
9:212 12760 03 FAP5 DC 2, 03
9:213 12761 @ DC 1, @

9:214

9:215 12764 $\bar{2}46$ DC 3, 246

9:216 12766 $\bar{0}8$ FAQ5 DC 2, 08
9:217 12767 @ DC 1, @

9:218

9:219

9:220 * EX 6 NO NORMALIZING REQUIRED---NO CARRY OUT -- SAME AS EX1

9:221

9:222 12769 $\bar{4}56777..63000@$ FAI7 DAC 50, EX7 NORMALIZING SHIFT RT - CARRY OUT- EXP ADJU

9:223

9:224 12870 $\bar{4}23$ DC 3, 423

9:225 12872 $\bar{0}4$ FAP7 DC 2, 04

9:226 12873 @ DC 1, @

9:227

9:228 12876 $\bar{7}45$ DC 3, 745

9:229 12878 $\bar{0}4$ FAQ7 DC 2, 04

9:230 12879 @ DC 1, @

9:231

9:232

9:233 12881 $\bar{4}56777..63000@$ FSI7 DAC 49, EX7 (SUB) NORM SHIFT RT - CARRY OUT- EXP ADJUS

9:234

9:235 * FSUB TEST - P IS NEGATIVE

9:236

9:237 12980 $\bar{4}2\bar{3}$ DC 3, -423

9:238 12982 $\bar{0}4$ FSP7 DC 2, 04

9:239 12983 @ DC 1, @

9:240

9:241 12986 $\bar{7}45$ DC 3, 745

9:242 12988 $\bar{0}4$ FSQ7 DC 2, 04

9:243 12989 @ DC 1, @

9:244

9:245

-

9:246	12991	456778..55460@	FAI8	DAC	50,EX8 NORM SHIFT RT-CARRY OT CAUSES EXP OV MACH
9:247					
9:248	13092	<u>4</u> 23		DC	3, 423
9:249	13094	<u>9</u> 9		FAP8	DC 2, 99
9:250	13095	@			DC 1, @
9:251					
9:252	13098	<u>7</u> 45		DC	3, 745
9:253	13100	<u>9</u> 9		FAQ8	DC 2, 99
9:254	13101	@			DC 1, @
9:255					
9:256	13103	<u>4</u> 56778..55460@	FSI8	DAC	50,EX8 (SUB) NORM SHIFT RT-CARRY OT- EXP OV MACH
9:257					
9:258			*	FSUB TEST - P IS NEGATIVE	
9:259					
9:260	13204	<u>4</u> 2 <u>3</u>		DC	3, -423
9:261	13206	<u>9</u> 9		FSP8	DC 2, 99
9:262	13207	@			DC 1, @
9:263					
9:264	13210	<u>7</u> 45		DC	3, 745
9:265	13212	<u>9</u> 9		FSQ8	DC 2, 99
9:266	13213	@			DC 1, @
9:267					
9:268	13215	<u>4</u> 56779..79790@	FAI9	DAC	50,EX9 ZERO FRACTION RESULT - MACHINE ZERO 0...0
9:269					
9:270	13316	<u>3</u> 45		DC	3, 345
9:271	13318	<u>0</u> 4		FAP9	DC 2, 04
9:272	13319	@			DC 1, @
9:273					
9:274	13322	<u>3</u> 4 <u>5</u>		DC	3, -345
9:275	13324	<u>0</u> 4		FAQ9	DC 2, 04
9:276	13325	@			DC 1, @

9:277
9:278

9:279 13327 $\bar{4}56771..44510@$ FAI10 DAC 50,EX10 NORMALIZING SHIFT LEFT - SHIFT LEFT- EXP
9:280

9:281 13428 $\bar{3}45$ DC 3, 345

9:282 13430 $\bar{0}4$ FAP10 DC 2, 04
9:283 13431 @ DC 1, @
9:284

9:285 13434 $\bar{3}2\bar{1}$ DC 3, -321

9:286 13436 $\bar{0}4$ FAQ10 DC 2, 04
9:287 13437 @ DC 1, @
9:288
9:289

9:290 13439 $\bar{4}56771..56000@$ FAI11 DAC 50,EX11 NORMALIZING SHIFT LEFT- EXP OV MACHINE ZP
9:291

9:292 13540 $\bar{3}45$ DC 3, 345

9:293 13542 $\bar{9}9$ FAP11 DC 2, -99
9:294 13543 @ DC 1, @
9:295

9:296 13546 $\bar{3}2\bar{1}$ DC 3, -321

9:297 13548 $\bar{9}9$ FAQ11 DC 2, -99
9:298 13549 @ DC 1, @
9:299
9:300 *
9:301
9:302 *FADD TEST DATA END -----
9:303
9:304
9:305 *FMUL TEST DATA START
9:306

9:307 13551 $\bar{3}33333..33330@$ FMULT DAC 45,===== FLOATING MULTIPLY (FMUL) TESTS ======@
9:308
9:309 * FMUL TESTS TABLE
9:310

9:311 13644 00000 FMINC DC 5, 0,, TABLE POINTER
 9:312
 9:313 * TABLE POINTER
 9:314 13659 00015 FMTP DS 15
 9:315
 9:316 * LIST OF TABLES
 9:317 13664 136941..413774 FMTBL DSA FMTBI, FMTBP, FMTBQ
 9:318
 9:319 13679 00005 FMI DS 5
 9:320 13684 00005 FMP DS 5
 9:321 13689 00005 FMQ DS 5
 9:322
 9:323
 9:324 * FMUL TEST IDENTIFICATION MESSAGES
 9:325 13694 142151..14871 FMTBI DSA FMI1, FMI10, FMI11, FMI2, FMI3, FMI4, FMI5
 9:326 13729 00000 DSA 00000
 9:327
 9:328 *MULTIPLICANDS
 9:329 13734 143181..214944 FMTBP DSA FMP1, FMP10, FMP11, FMP2, FMP3, FMP4, FMP5
 9:330 13769 00000 DSA 00000
 9:331
 9:332 *MULTIPLIERS
 9:333 13774 143241..814950 FMTBQ DSA FMQ1, FMQ10, FMQ11, FMQ2, FMQ3, FMQ4, FMQ5
 9:334 13809 00000 DSA 00000
 9:335
 9:336
 9:337 * FMUL DATA SYMBOLS
 9:338
 9:339 13859 000000..000000 DC 50, 0
 9:340 13860 000000..000000 DSC 50, 0
 9:341 13911 00 FMPRD DC 2, 0
 9:342 13912 @ DC 1, @
 9:343
 9:344
 9:345 * REF 227-5630-1 IBM 1620 FLOATING POINT FEATURE CE MANUAL

9:379

9:380	14428	<u>345</u>	DC	3, -345
9:381	14430	<u>07</u>	FMP10 DC	2, -07
9:382	14431	@	DC	1, @
9:383				
9:384	14434	<u>432</u>	DC	3, 432
9:385	14436	<u>03</u>	FMQ10 DC	2, 03
9:386	14437	@	DC	1, @
9:387				
9:388				
9:389	14439	<u>456771..04000@</u>	FMI11 DAC	50, EX1C NO EXP MOD REQUIRED -MP,-MQ, -EP,-EQ (FMI)
9:390				
9:391	14540	<u>345</u>	DC	3, -345
9:392	14542	<u>07</u>	FMP11 DC	2, -07
9:393	14543	@	DC	1, @
9:394				
9:395	14546	<u>432</u>	DC	3, -432
9:396	14548	<u>03</u>	FMQ11 DC	2, -03
9:397	14549	@	DC	1, @
9:398				
9:399				
9:400			*	FMUL EXAMPLE 2. EXPONENT MODIFY REQUIRED
9:401			*	
9:402			*	123456789 123456789 123456789 123456789 123456789 123456789
9:403	14551	<u>456772..04000@</u>	FMI2 DAC	49, EX2 EXPONENT MOD REQ-ED +MP,+MQ, +EP,+EQ (FMUL)
9:404				
9:405	14650	<u>123</u>	DC	3, 123
9:406	14652	<u>02</u>	FMP2 DC	2, 02
9:407	14653	@	DC	1, @
9:408				
9:409	14656	<u>246</u>	DC	3, 246
		—		

```

9:410 14658 04          FMQ2 DC    2, 04
9:411 14659 @           DC    1, @
9:412
9:413
9:414 * FMUL EXAMPLE 3. DIGIT FORCE -- MACHINE INFINITY
9:415
9:416 *               123456789 123456789 123456789 123456789 1234
9:417 14661 456773..04000@ FMI3 DAC 44,EX3 DIGIT FORCE -- MACHINE INFINITY (FMUL) @
9:418

9:419 14750 345          DC    3, 345
9:420 14752 43            FMP3 DC    2, 43
9:421 14753 @             DC    1, @
9:422

9:423 14756 432          DC    3, 432
9:424 14758 64            FMQ3 DC    2, 64
9:425 14759 @             DC    1, @
9:426
9:427
9:428 * FMUL EXAMPLE 4. EXPONENT OVERFLOW -- MODIFY PE TO 99
9:429
9:430 *               123456789 123456789 123456789 123456789 123456
9:431 14761 456774..04000@ FMI4 DAC 49,EX4 EXPONENT OVERFLOW -- MODIFY PE TO 99 (FMUL)
9:432

9:433 14860 123          DC    3, 123
9:434 14862 38            FMP4 DC    2, 38
9:435 14863 @             DC    1, @
9:436

9:437 14866 246          DC    3, 246
9:438 14868 62            FMQ4 DC    2, 62
9:439 14869 @             DC    1, @
9:440
9:441
9:442 * FMUL EXAMPLE 5. SPECIAL CASE 999 X 199
9:443
9:444 *               123456789 123456789 123456789 123456789 12345

```

9:445 14871 $\bar{4}56775..04000@$ FMI5 DAC 35,EX5 SPECIAL CASE 999 X 199 (FMUL) @
9:446

9:447 14942 $\bar{9}99$ DC 3, 999

9:448 14944 $\bar{0}1$ FMP5 DC 2, 01
9:449 14945 @ DC 1, @
9:450

9:451 14948 $\bar{1}99$ DC 3, 199

9:452 14950 $\bar{0}1$ FMQ5 DC 2, 01
9:453 14951 @ DC 1, @
9:454
9:455
9:456 *FMUL TEST DATA END -----
9:457
9:458 * CONSTANTS AND STATUS CODES

9:459 14952 $\bar{1}$ TRUE DC 1,1

9:460 14953 $\bar{0}$ FALSE DC 1,0
9:461

9:462 14954 $\bar{0}$ ESTAT DC 1,0,,, FL COMMAND TEST ERR RTN STATUS

9:463 14955 $\bar{0}$ STAT DC 1,0,,, FL COMMAND TEST STATUS
9:464

9:465 14957 $\bar{3}33333..33330@$ FAIL DAC 17,==== FAILED ====@

9:466 14991 $\bar{2}02057..20200@$ PASS DAC 11,--PASSED--@
9:467
9:468

9:469 15012 $\bar{0}$ OV DC 1,0,,, OVERFLOW FLAG

9:470 15013 $\bar{0}$ XCK DC 1,0,,, EXPONENT OVERFLOW FLAG
9:471

9:472 15015 $\bar{0}05665..66000@$ OVM1 DAC 11, OVERFLOW @

9:473 15037 $\bar{0}04567..66000@$ XCKM1 DAC 20, EXPONENT OVERFLOW @
9:474

9:475

9:476

9:477

9:478

9:479 00402

* -----
* -----
* -----
DEND FPMAIN

Symbol Cross-Reference Table

Id Source File

---	---
1	fp10-main.sps
2	fp20-tftest.sps
3	fp21-tftest.sps
4	fp30-fadd.sps
5	fp31-fixadd.sps
6	fp40-fmul.sps
7	fp41-fixmul.sps
8	fp50-subroutines.sps
9	fp60-symbols-dend.sps

Symbol Addr. Type Defined References

-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
A21	03216	<inst>	5:45	4:101	5:49	5:58	5:66	5:74	5:305	5
A21A1	06253	DC	5:388	5:62	5:90	5:102	5:106	5:113		
A21A2	06258	DC	5:389	5:70	5:91	5:103	5:107			
A21D1	06360	DC	5:393	5:77	5:81	5:301				
A21RC	06366	DC	5:396	5:301	5:308					
A9	05932	DC	5:366	5:272	5:275					
ACMF1	04704	<inst>	5:238	5:229	5:230					
ACMF2	04752	<inst>	5:242	5:231	5:232					
ACO	04596	<inst>	5:229	5:222						
ADDM	04104	<inst>	5:165	5:122	5:142	5:161				
ADIGT	04404	<inst>	5:204	5:208						
ADJP	03780	<inst>	5:129	5:123						
ADJP1	03840	<inst>	5:135	5:130						
ADJQ	03936	<inst>	5:147	5:124						
ADJQ1	04008	<inst>	5:154	5:149						
ADSA	03118	DSA	4:102	4:16						
AED	05281	DC	5:320	5:95	5:96	5:97	5:98	5:129	5:136	5
				5:155						
AELS	05088	<inst>	5:284	5:259						
AEQVM	05667	DAC	5:356							
AEP	05403	DC	5:332	5:88	5:90	5:96	5:121	5:151	5:255	5
AEPOQ	05619	DC	5:348	5:132	5:151	5:223	5:254	5:255	5:258	5
				5:265	5:284	5:285	5:287	5:293	5:293	5
AEQ	05511	DC	5:340	5:89	5:91	5:97	5:121	5:132	5:256	5
AFLAG	04368	<inst>	5:200	5:201						
AFLG	05286	DC	5:322	5:197	5:206	5:214				

AINF	04968	<inst>	5:270	5:264						
AINF1	05028	<inst>	5:275	5:271						
AINF2	05064	<inst>	5:278	5:270						
AL	03696	<inst>	5:115	5:117						
ALDON	01146	<inst>	1:154							
AMKFP	05148	<inst>	5:293	5:133	5:152	5:224	5:261	5:266	5:279	5
AML	05276	DC	5:319	5:112	5:116	5:129	5:139	5:148	5:158	5
				5:235	5:237	5:239	5:241	5:276		
AMP	05397	DSC	5:329	5:104	5:106	5:135	5:137	5:137	5:138	5
				5:141	5:150	5:169	5:229	5:230	5:231	5
				5:238	5:238	5:244				
AMPA	05296	DC	5:325	5:113	5:115	5:117	5:198	5:200	5:201	5
				5:204	5:207	5:211				
AMPAA	06248	DC	5:386							
AMPQ	05613	DSC	5:345	5:131	5:150	5:168	5:169	5:170	5:184	5
				5:193	5:198	5:210	5:212	5:213	5:215	5
				5:243	5:244	5:245	5:246	5:246	5:247	5
				5:270	5:271	5:272	5:273	5:275	5:278	5
				5:295						
AMQ	05505	DSC	5:337	5:105	5:107	5:131	5:154	5:156	5:156	5
				5:160	5:160	5:170	5:242	5:242	5:245	5
ANRM	05622	DC	5:352	5:166	5:187	5:196	5:259			
AOV	05621	DC	5:351	5:167	5:175	5:233	5:260			
AOVM	05625	DAC	5:355							
APQFP	05829	DC	5:359							
ASF	04464	<inst>	5:210	5:204						
ASF2	04524	<inst>	5:215	5:210						
ASHFT	04212	<inst>	5:183	5:174						
ASTBL	11728	DS	9:111	4:42	4:43	4:44	4:52	4:53	4:54	4
				4:61	4:64	4:65	4:69	4:70	4:71	
AWK1	06034	DC	5:371	5:195	5:212	5:215				
AZ100	06142	DSC	5:379							
AZ101	06243	DSC	5:383	5:104	5:105	5:168	5:184	5:195	5:213	5
				5:243	5:288					
AZ2	06037	DC	5:374							
AZ5	06042	DC	5:375	5:88	5:89	5:95	5:254			
AZCK	04548	<inst>	5:221	5:176	5:193					
AZCNT	05291	DC	5:323	5:188	5:205	5:284				
AZRO	04572	<inst>	5:223	5:185						
BTSUB	02392	<inst>	3:18	2:131	3:24					
CF	09922	<inst>	8:188	3:23	5:300	7:136	8:191	8:201	8:230	8
CFA1	10310	DC	8:239	8:194	8:198	8:215	8:220	8:224		
CFA2	10315	DC	8:240	8:204	8:208	8:215	8:221	8:224		
CFINI	10126	<inst>	8:212	8:205						
CFRC	10316	DC	8:241	8:212	8:217	8:226	8:233			

CFRT	10234	<inst>	8:228	8:225							
CFSEC	10030	<inst>	8:201	8:195							
DONE	10389	DAC	9:9	1:157							
EA1	09914	DC	8:159	8:113	8:117						
ERPAS	09662	<inst>	8:126	8:117							
ERPSW	09734	<inst>	8:135	8:130							
ERRC	09915	DC	8:160	8:115	8:131	8:153					
ERRT	09842	<inst>	8:148	8:132							
ERSW2	09686	<inst>	8:129	8:122	8:124						
ERSW3	09626	<inst>	8:122	8:119							
ERTN	09518	<inst>	8:106	2:141	4:107	6:74	8:110	8:150	8:156		
ESTAT	14954	DC	9:462	2:142	2:144	4:108	4:110	6:75	6:77		
EXADJ	04824	<inst>	5:254	5:216	5:234						
FACR	05260	<inst>	5:312								
FAEND	03186	<inst>	4:113	4:110							
FAI	11763	DS	9:120	4:69	4:76						
FAI1	12209	DAC	9:165	9:125	9:139						
FAI10	13327	DAC	9:279	9:125	9:139						
FAI11	13439	DAC	9:290	9:125	9:139						
FAI2	12321	DAC	9:176	9:125	9:139						
FAI3	12433	DAC	9:187	9:125	9:139						
FAI4	12545	DAC	9:198	9:125	9:139						
FAI5	12657	DAC	9:209	9:125	9:139						
FAI7	12769	DAC	9:222	9:125							
FAI8	12991	DAC	9:246	9:125							
FAI9	13215	DAC	9:268	9:125	9:139						
FAIL	14957	DAC	9:465	8:120							
FAINC	11713	DC	9:108	4:32	4:60	4:64	4:65	4:67			
FALSE	14953	DC	9:460	2:65	5:166	5:167	5:187	8:115	8:217		8
FANXT	02826	<inst>	4:57	4:45	4:115						
FAP	11768	DS	9:121	4:70	4:82	4:83	4:84	4:85	4:102		
FAP1	12312	DC	9:168	9:129	9:143						
FAP10	13430	DC	9:282	9:129	9:143						
FAP11	13542	DC	9:293	9:129	9:143						
FAP2	12424	DC	9:179	9:129	9:143						
FAP3	12536	DC	9:190	9:129	9:143						
FAP4	12648	DC	9:201	9:129	9:143						
FAP5	12760	DC	9:212	9:129	9:143						
FAP7	12872	DC	9:225	9:129							
FAP8	13094	DC	9:249	9:129							
FAP9	13318	DC	9:271	9:129	9:143						
FAQ	11773	DS	9:122	4:71	4:89	4:102					
FAQ1	12318	DC	9:172	9:133	9:147						
FAQ10	13436	DC	9:286	9:133	9:147						
FAQ11	13548	DC	9:297	9:133	9:147						

FAQ2	12430	DC	9:183	9:133	9:147				
FAQ3	12542	DC	9:194	9:133	9:147				
FAQ4	12654	DC	9:205	9:133	9:147				
FAQ5	12766	DC	9:216	9:133	9:147				
FAQ7	12878	DC	9:229	9:133					
FAQ8	13100	DC	9:253	9:133					
FAQ9	13324	DC	9:275	9:133	9:147				
FARIT	03042	<inst>	4:89	4:12	4:34				
FARPT	02946	<inst>	4:74	4:111					
FASUM	12205	DC	9:156	4:82	4:84	4:89	4:102		
FATBI	11778	DSA	9:125	4:39	9:114				
FATBL	11733	DSA	9:114	4:39	4:40	4:41	4:42	4:43	4:44
FATBP	11833	DSA	9:129	4:40	9:114				
FATBQ	11888	DSA	9:133	4:41	9:114				
FATST	02466	<inst>	4:7	1:137	1:144	4:10	4:14	4:18	4:62
FIRST	00462	<inst>	1:111						
FMCR	07720	<inst>	7:145						
FMEND	06870	<inst>	6:80	6:77					
FMI	13679	DS	9:319	6:40	6:47				
FMI0	13915	DAC	9:349						
FMI1	14215	DAC	9:367	9:325					
FMI10	14327	DAC	9:378	9:325					
FMI11	14439	DAC	9:389	9:325					
FMI2	14551	DAC	9:403	9:325					
FMI3	14661	DAC	9:417	9:325					
FMI4	14761	DAC	9:431	9:325					
FMI5	14871	DAC	9:445	9:325					
FMINC	13644	DC	9:311	6:18	6:31	6:35	6:36	6:38	
FML	07748	DC	7:151	7:77	7:81	7:92	7:108	7:109	7:114
FMNXT	06516	<inst>	6:29	6:81					
FMP	13684	DS	9:320	6:41	6:53	6:54	6:55	6:56	6:69
FMP0	14109	DC	9:353						
FMP1	14318	DC	9:370	9:329					
FMP10	14430	DC	9:381	9:329					
FMP11	14542	DC	9:392	9:329					
FMP2	14652	DC	9:406	9:329					
FMP3	14752	DC	9:420	9:329					
FMP4	14862	DC	9:434	9:329					
FMP5	14944	DC	9:448	9:329					
FMPRD	13911	DC	9:341	6:53	6:55	6:58	6:69		
FMQ	13689	DS	9:321	6:42	6:58	6:69			
FMQ0	14212	DC	9:360						
FMQ1	14324	DC	9:374	9:333					
FMQ10	14436	DC	9:385	9:333					
FMQ11	14548	DC	9:396	9:333					

ON	09153	DAC	8:70	8:45	8:50	8:55	8:60				
OV	15012	DC	9:469								
OVM1	15015	DAC	9:472	4:94	6:62						
PAR1	08483	DC	7:194	7:107	7:108	7:109	7:110	7:111	7:113		
PAREA	08478	DC	7:193	7:101	7:121	7:123					
PARW	08488	DC	7:195	7:111	7:112	7:113	7:121	7:122	7:123		7
PASS	14991	DAC	9:466	8:127							
PQFP	08153	DC	7:170	5:294	5:295	5:301	7:131	7:132	7:137		
PQMA	08050	DSC	7:165	7:95	7:126	7:132					
PRSH	07548	<inst>	7:121	7:110	7:117						
RETAD	02460	DS	3:35	2:130	3:31						
SETS1	08614	<inst>	8:6	1:113	8:16						
STAT	14955	DC	9:463	2:65	2:135	2:142	3:24	4:102	4:108		6
				6:75							
SUBTB	02754	<inst>	4:49	4:35							
SUMEX	08159	DC	7:173	7:65	7:66	7:67	7:86	7:90	7:98		7
				7:118	7:130	7:130	7:131				
SWMS1	09105	DAC	8:69	8:39							
SWMS2	09185	DAC	8:73	8:28							
SWMS3	09245	DAC	8:74	8:31							
SWMS4	09291	DAC	8:75	8:34							
SWMS5	09341	DAC	8:76	8:12	8:143						
SWMS6	09427	DAC	8:77	8:141							
SWST	09065	DAC	8:67	8:37							
SWSTA	08704	<inst>	8:26	8:8	8:64	8:135					
TBTFL	02112	<inst>	2:126	2:106							
TER	02220	<inst>	2:141	2:81	2:101	2:122	2:134				
TEXCK	02184	<inst>	2:133	2:71	2:80	2:100	2:121				
TFD	10797	DS	9:42	2:53	2:71	2:76	2:77	2:91	2:95		2
				2:110	2:112	2:118	2:126	2:127			
TFD1	11035	DC	9:60	9:50							
TFD2	11237	DC	9:67	9:50							
TFD3	11375	DC	9:74	9:50							
TFD4	11537	DC	9:82	9:50							
TFD5	11707	DC	9:92	9:50							
TFEND	02266	<inst>	2:147	2:144							
TFI	10792	DS	9:41	2:52	2:57						
TFI1	10869	DAC	9:56	9:46							
TFI2	11039	DAC	9:63	9:46							
TFI3	11241	DAC	9:70	9:46							
TFI4	11379	DAC	9:78	9:46							
TFI5	11541	DAC	9:88	9:46							
TFINC	10767	DC	9:32	2:30	2:44	2:48	2:50				
TFINS	01284	<inst>	2:13	2:12	2:74	2:89	2:105				
TFNXT	01440	<inst>	2:40	2:149							

TFRPT	01536	<inst>	2:56	2:145							
TFSL	01740	<inst>	2:89	2:75							
TFSLC	01860	<inst>	2:99	2:95	2:96	2:97	2:97				
TFSR	01896	<inst>	2:105	2:90							
TFSRC	02076	<inst>	2:120	2:118	2:119						
TFSRF	02040	<inst>	2:117	2:114							
TFTBD	10837	DSA	9:50	2:35	9:38						
TFTBI	10802	DSA	9:46	2:34	9:38						
TFTBL	10782	DSA	9:38	2:34	2:35	2:36	2:37				
TFTBP	10777	DS	9:35	2:36	2:37	2:44	2:45	2:48	2:52	2	
TFTST	01236	<inst>	2:7	1:116	1:121	1:126	1:131	2:10	2:16	2	
TFWT	01356	<inst>	2:24	2:18							
TITLE	10319	DAC	9:8	1:102							
TRUE	14952	DC	9:459	2:135	5:175	5:196	8:131	8:212			
WK1	10658	DC	9:21	2:66	2:67	2:76	2:79	2:93	2:94	2:116	2
				2:98	2:99	2:109	2:111	2:115			
				2:117	2:120	2:129	2:133	3:24			
WK1Z	10554	DSC	9:16	2:66	2:68						
WK2	10761	DC	9:26	2:68	2:69						
WTITL	02610	<inst>	4:26	4:20							
XCK	15013	DC	9:470								
XCKM1	15037	DAC	9:473	4:96	6:64						

CU03– Indirect Addressing Diagnostic


```
//=====
//  
// CU03 - Indirect Addressing Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Bypass error type out  
// OFF - Type out routine number on error  
// PS2: ON - Loop in routine  
// OFF - Continue to next routine  
// PS3: ON - Stop on error  
// OFF - Do not stop on error, continue  
// PS4: ON - Repeat test CU03  
// OFF - Run test CU03 once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - STOP  
//  
// Start addresses:  
//  
// 00828 - Full test  
//  
// Directions:  
//  
// 1. Load CU03 diagnostic  
// 2. Press START  
// 3. Press START  
//=====
```


Sample Output - CU03

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR INDIRECT ADDRESSING TEST. THEN START.

START ROUTINES. ETOS FOLLOW.

TEST ROUTINES COMPLETED. IF SW1 OFF, NO ROUTINE NOS TYPED OUT, AND NO HANG-UPS,
MACHINE PERFORMED TESTS PROPERLY.

NO. 2125574
SHEET 0
OF 27

DIAGNOSTIC TEST

QITLE 1620 INDIRECT ADDRESSING DIAGNOSTIC TEST - CU03
MACH.TYPE 1620 BY J.H.M. APPR. GIA DATE 4-11-62

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404568	12-15-60	1-27
404618	5-15-61	1, 1A, 4, 5A, 10, 14, 16, 20, 27
404675	4-11-62	2, 3, 4, 5A, 8, 13, 14, 16, 23, 26, 27

E/C NO.	404568	404618	404675				
DATE	12-15-60	5-15-61	4-11-62				

INDIRECT ADDRESSING**A. SCOPE:**

This test is essentially a fault detection test designed to check the indirect addressing circuits. Both the P and Q fields are used as indirect addresses and specify data fields in both even and odd memory positions. Chaining of indirect addresses in both the P and Q fields is checked for two addresses.

All immediate operation codes, write alphabetic, branch, branch indicate, and branch no indicate codes are checked to determine that a flag in the low order position of the Q field will not indicate an indirect address.

The control code is checked to determine that the P field is not indirect addressable, and both the P and Q fields for branch back, NOP, and halt are checked for being non-indirect addressable.

B. SET UP:

Data Switches should normally be set to STOP. This will cause the program to stop at the end of the cycle on which a parity error occurs.

The four console switches should be set as desired. The normal setting is, all switches OFF. These switches have the following functions in this test:

SWITCH #1	ON - Bypass error type out OFF - Type out routine number on error
SWITCH #2	ON - Loop in routine OFF - Continue to next routine
SWITCH #3	ON - Stop on error OFF - Do not stop on error; continue
SWITCH #4	ON - Repeat test CU03

NORMAL LOAD FROM PAPER TAPE READER:

To run the entire test, load the tape in the reader, put in REEL mode, and READY the reader. Reset the 1620. Insert, key in the instructions:
360009600300
4900828. (MAR should read 00018) Release and start, then follow instructions typed out.

NORMAL LOAD FROM CARD I/O:

Place card deck in the reader. Reset 1620. Depress load key located on the card reader. The first card is read into the buffer and its data automatically transferred into the first 80 positions of core storage. Following the transfer of data into the 80 low order positions of core storage, the computer will simulate a release and program start at location 00000.

TO PRODUCE A NEW PAPER TAPE:

To regenerate, or produce another tape for input, read the Master tape into the last part of memory and then dump memory to the paper tape punch. The instructions required to load and dump memory to produce another paper tape are:

36 14444 00300
35 14444 00200
48

C. TEST METHOD:

The test is composed of sub-routines, each of which is designed to check some aspect of the Indirect Addressing feature.

A failure of a sub-routine to perform the indirect addressing properly will initiate an error routine. There are three different formats for the error routines; timeouts, halts, and "hang-ups" in indirect addressing cycles.

The normal run of this test performs each routine 100 times before testing sense switch #4. If sense switch #4 is off, the test completed routine is executed and the halt operation code is tested to determine that the P and Q fields are not indirect addressable.

The first sub-routine executed types the setting of the sense switches, the name of the test, and instructions to follow.

Routines 002-005 check for P field indirect addressing on the immediate operation codes add and subtract. If the indirect address defines the field of the augend or minuend, a certain answer is obtained. The result of the arithmetic operation is compared against the predetermined answer and equal comparison indicates that the indirect address operation was correct. If the augend or minuend is not defined by the indirect address, a different answer results. If the first comparison is unequal, the results of the arithmetic operation is compared against the predetermined error result. (The result if indirect addressing is not executed). If this comparison is equal, an error routine is executed and the routine number typed out. If the comparison is unequal, the program halts. Displaying IR-1 will indicate the routine number in which the program halts.

Routines 006-008 check for the chaining of two indirect addresses in the Q field of a multiply operation. The product is compared against a predetermined answer. An equal comparison indicates that the chaining of indirect addresses functioned properly. If this comparison is unequal, another comparison is made against a predetermined answer that would result in the first indirect address was not recognized. An equal comparison on this second compare initiates the execution of an error routine that types the routine number;

while an unequal comparison on the second compare causes the program to halt.

Routines 009 and 010 check for the chaining of two indirect addresses in the P field of a transmit field operation. The transmitted field is compared against predetermined results. An equal comparison indicates that the chaining operation was performed correctly, and the program branches to the next routine. An unequal comparison on the first compare initiates a second compare against a result that would occur if the first indirect address was ignored. An equal comparison branches to an error routine that types out the routine number; while an unequal comparison halts the program.

Routines 011-016 check that a flag bit on the low order position of the Q field of the immediate operations add, subtract, multiply, compare, transmit digit, and transmit field does not define an indirect address. The routine numbers will type out as an error indication if the Q field is recognized as an indirect address.

Routine 017 checks that the Q field of the write alphabetic operation will not recognize an indirect address. A T should be typed for each routine pass. If the Q field is recognized as an indirect address, the program will "hang-up".

Routine 018 checks that the Q field of a transmit record can be an indirect address. A comparison is made to determine if the indirect address was used or ignored. An equal comparison advances the program to the next routine. An unequal comparison advances the program to the error routine that types out the routine number and the record transmitted.

Routines 019 and 519 check that the Q field of the branch and transmit immediate operation and both the P and Q field of the branch back operation will not recognize an indirect address. The routine number is typed out if the incorrect field is transmitted on the branch and transmit immediate operation. If the P and Q fields of the branch back instruction recognize indirect addresses, the program will "hang-up" in indirect addressing cycles on the branch back instruction.

Routines 020-022 check that the Q field of a branch indicate, branch no indicate, and branch operation will not recognize an indirect address. If the Q fields of these operations recognize an indirect address, the addresses are such that the program will "hang-up" in indirect address cycles on the "op code" being checked.

Routine 023 checks that on NOP the P and Q fields will not recognize indirect addresses. The indirect address in the P field is the Q field address, and the indirect address in the Q field is the P field address. If an indirect address is recognized, a loop will be set up and the program will "hang-up" in indirect address cycles.

Routine 024 checks that the P field of a control operation does not recognize an indirect address. A "hang-up" in indirect address cycles will result if an indirect address is recognized.

Routine 025 is the times 100 routine, and routine 026 is the test completed routine and also checks that on a halt operation neither the P nor the Q fields will recognize an indirect address.

The complete normal typeout information will be as follows:

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR INDIRECT
ADDRESSING TEST. THEN START.
START ROUTINES. ETOS FOLLOW.

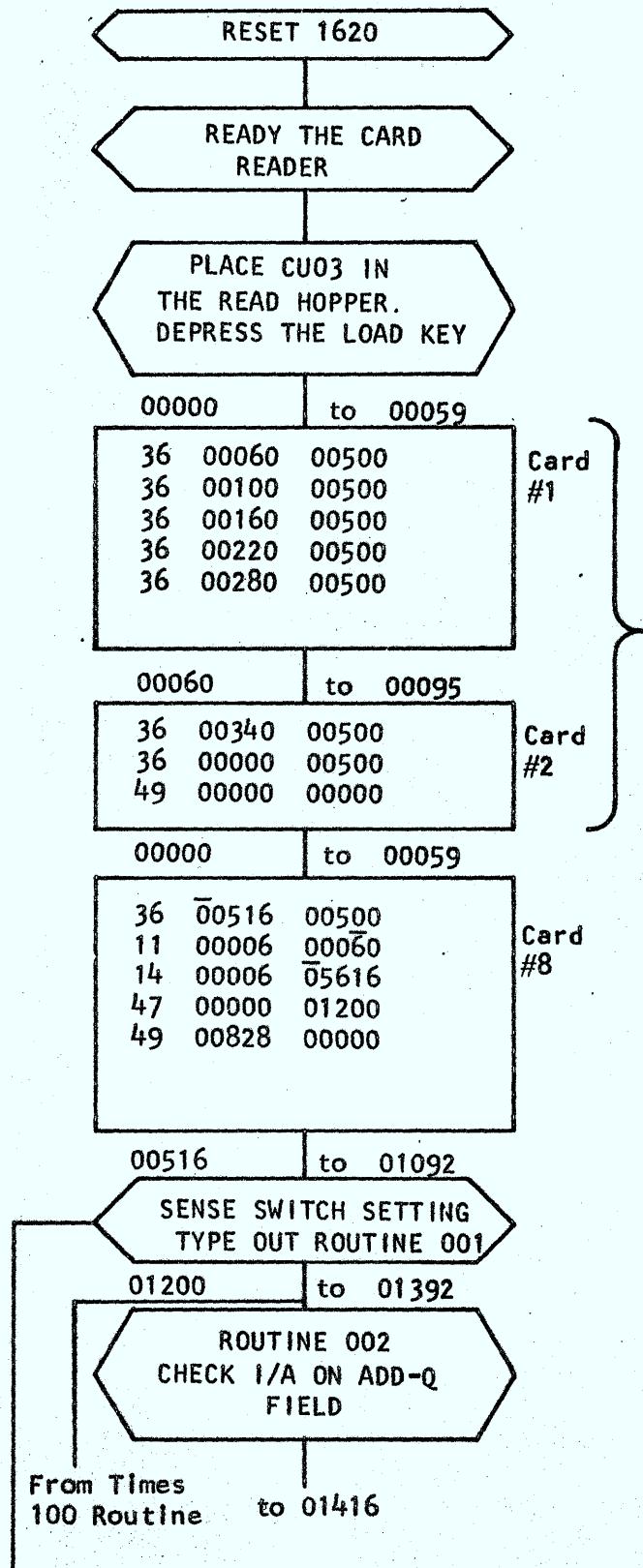
START ROUTINES. ETCOS FOLLOW.

TEST ROUTINES COMPLETED. IF SW1 OFF, NO ROUTINE NOS TYPED OUT, AND NO HANG-UPS, MACHINE PERFORMED TESTS PROPERLY.

PN 2125574
EC 404675

CU03 FLOW CHART
WITH 1622 I/O

Page 5A



First and second Load Cards load the math tables and Program Load Card.
(Cards 3 through 7 contain the math tables.)

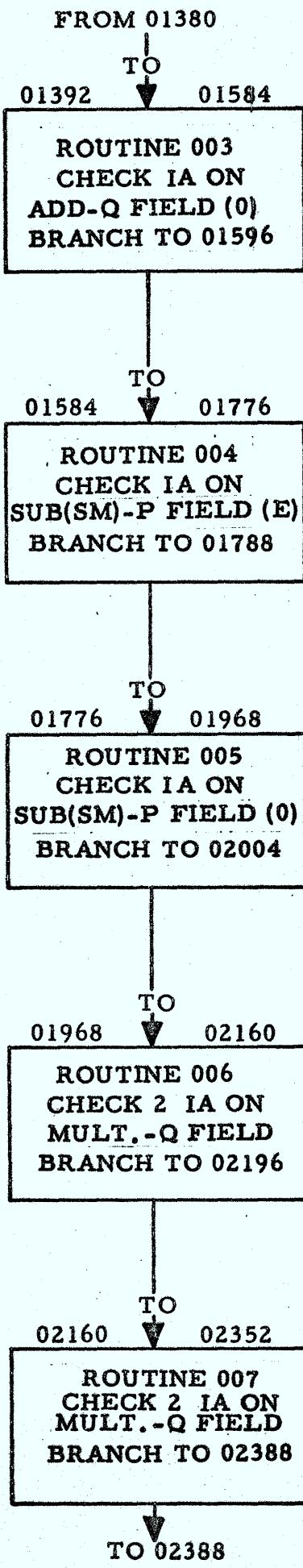
Eighth Load Card contains instructions for loading core storage.

This routine will indicate if B1 and/or BN1 are working. The 1620 will halt with 01091 in MAR

If Add is {
077
077
154
077
210
287} the I/A was correct
If Add is {
210
287} the I/A was not initiated

PN 2125574
EC 404675

- 1) SW 4 On (05616)
- 2) Start After Halt (05640)



IF THE ADD IS 099 THE I/A
099 WAS CORRECT
198

IF THE ADD IS 403 THE I/A WAS
099 NOT INITIATED
502

IF THE SUBT. IS 099 THE I/A WAS
-066 CORRECT
033

IF THE SUBT. IS 592 THE I/A WAS
-066 NOT INITIATED
526

IF THE SUBT. IS 099 THE I/A WAS
-077 CORRECT
022

IF THE SUBT. IS 783 THE I/A WAS
-077 NOT INITIATED
706

IF THE MULTIPLY IS 666 THE I/A WAS
x99 CORRECT
65934

IF THE MULTIPLY IS 666 THE FIRST I/A
x72 WAS NOT
47952 INITIATED

IF PRODUCT IS 50616, 2ND I/A WAS NOT
INITIATED

IF THE MULTIPLY IS 888 THE I/A WAS
x67 CORRECT
68376

IF PRODUCT IS 57720, THE I/A WAS NOT
INITIATED

IF PRODUCT IS 59496, THE 2ND I/A WAS
NOT INITIATED

FROM 02340

Page -7-

02352

02544

ROUTINE 008
CHECK 2 IA ON
MULT-Q FIELD
BRANCH TO 02580

IF MULTIPLY IS $\bar{4}44$ THE IA WAS
 $\times \bar{3}3$ CORRECT
 $\bar{1}4652$

02544

02772

ROUTINE 009
CHECK 2 IA ON
TF - P FIELD
BRANCH TO 02808

IF FIELD 02562-02566 IS $\bar{7}0248$, I/A WAS
CORRECT. IF FIELD 02556-02560 IS
 $\bar{7}0248$, FIRST I/A WAS NOT INITIATED.
IF FIELD 02544-02548 IS $\bar{7}0248$, 2ND I/A
WAS NOT INITIATED.

02772

03000

ROUTINE 010
CHECK 2 IA ON
TF - P FIELD
BRANCH TO 03036

IF FIELD 02791-02795 IS $\bar{8}2047$, I/A
WAS CORRECT. IF FIELD 02785-02789
IS $\bar{8}2047$, I/A WAS NOT INITIATED. IF
FIELD 02773-02777 IS $\bar{8}2047$, 2ND I/A
WAS NOT INITIATED.

03000

03216

ROUTINE 011
CHECK FOR NO IA
ON AM-Q FIELD
BRANCH TO 03240

I/A SHOULD BE INHIBITED. IF SUM IS
 $\bar{5}877$, NO I/A. IF SUM IS $\bar{9}110$, I/A WAS
NOT INHIBITED.

03216

03420

ROUTINE 012
CHECK FOR NO IA
ON SM-Q FIELD
BRANCH TO 03444

I/A SHOULD BE INHIBITED. IF REMAINDER
IS $\bar{8}004$, NO I/A. IF REMAINDER IS
 $\bar{7}333$, I/A WAS NOT INHIBITED.

TO 03444

PN 2125574
EC 404568

FROM 03408

TO

03420

03612

ROUTINE 013
CHECK FOR NO IA
ON MM-Q FIELD
BRANCH TO 03672

TO

03660

03804

ROUTINE 014
CHECK FOR NO IA
ON CM-Q FIELD
BRANCH TO 03816

TO

03804

03960

ROUTINE 015
CHECK FOR NO IA
ON TDM-Q FIELD
BRANCH TO 03984

TO

03960

04116

ROUTINE 016
CHECK FOR NO IA
ON TFM-Q FIELD
BRANCH TO 04272

I

TO

04116

04332

ROUTINE 017
CHECK FOR NO IA
ON WA-Q FIELD
BRANCH TO 04356

TO 04356

Page -8-

IF PRODUCT IS 3441, I/A WAS
INHIBITED. IF PRODUCT IS 2442,
I/A WAS NOT INHIBITED.

IF E/Z AFTER COMPARE, I/A WAS
INHIBITED. IF ERROR TYPE OUT,
COMPARE UNEQUAL.

IF TRANSMITTED DIGIT IS 5, NO I/A.
IF TRANSMITTED DIGIT IS 9, I/A WAS
NOT INHIBITED.

IF TRANSMITTED FIELD IS 03971,
NO I/A. IF TRANSMITTED FIELD IS
1287, I/A WAS NOT INHIBITED.

A "T" SHOULD BE TYPED FOR EACH ROUTINE
PASS. A "HANG-UP" INDICATES I/A IS NOT
BEING INHIBITED.

PN 2125574
EC 404675

FROM 04296

TO

04332 ▼ 04500

ROUTINE 018
CHECK FOR IA
ON TR - Q FIELD
BRANCH TO 04512

TO

04500 ▼ 04632

ROUTINE 019
CHECK FOR NO IA
ON BTM - Q FIELD
BRANCH TO 04740

04632 TO 04716

ROUTINE 519
CHECK FOR NO IA
ON BB

04716 ▼ 04896

ROUTINE 020
CHECK FOR NO IA
ON B1 - Q FIELD
BRANCH TO 04908

04896 ▼ 05064

ROUTINE 021
CHECK FOR NO IA
ON BN1 - Q FIELD
BRANCH TO 05064

TO 05064

Page -9-

IF TRANSMITTED RECORD IS 968,
I/A WAS EXECUTED. IF TRANSMITTED
RECORD IS 128, NO I/A.

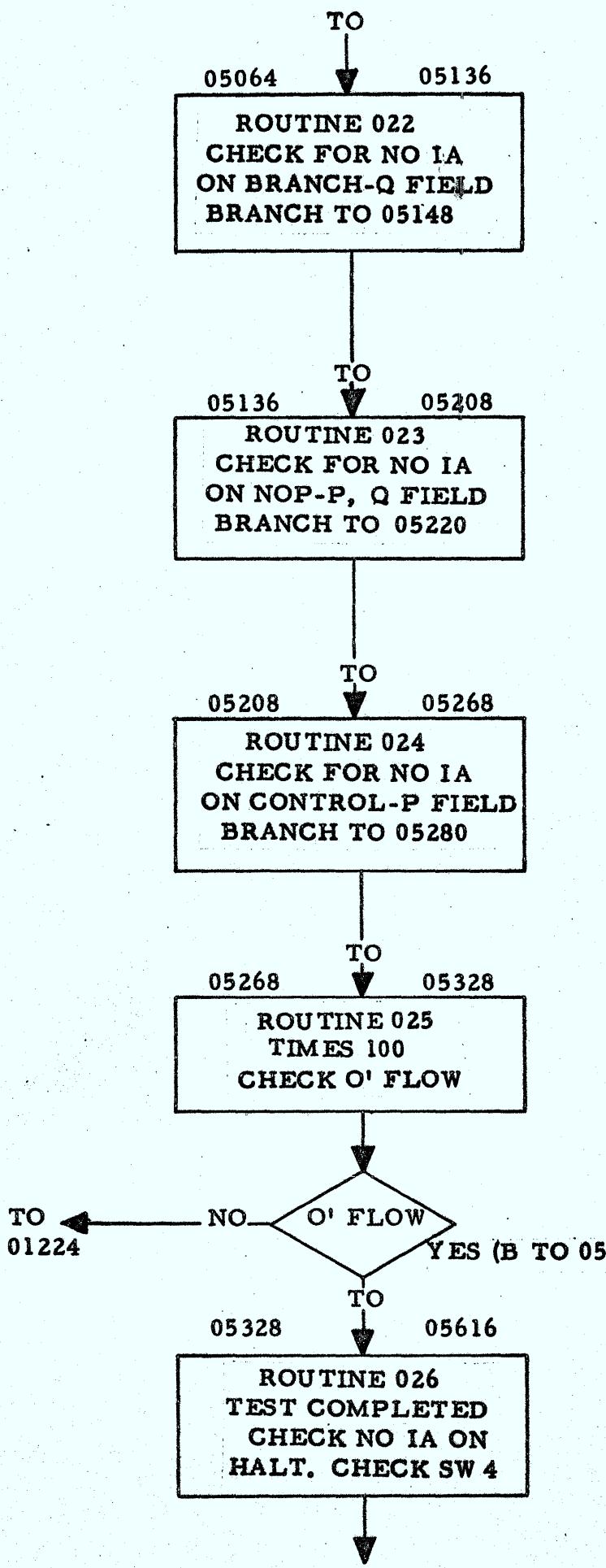
IF TRANSMITTED FIELD IS 04511,
NO I/A. IF TRANSMITTED FIELD IS
98765, I/A WAS EXECUTED.

A "HANG-UP" IN 1 TIME WITH A 42
OP CODE AT INSTRUCTION 04680
INDICATES IA IS NOT BEING
INHIBITED.

A "HANG-UP" IN 1 TIME WITH A 46
OP CODE AT INSTRUCTION 04764
INDICATES IA IS NOT BEING
INHIBITED.

A "HANG-UP" IN 1 TIME WITH A 47
OP CODE AT INSTRUCTION 04932
INDICATES IA IS NOT BEING
INHIBITED.

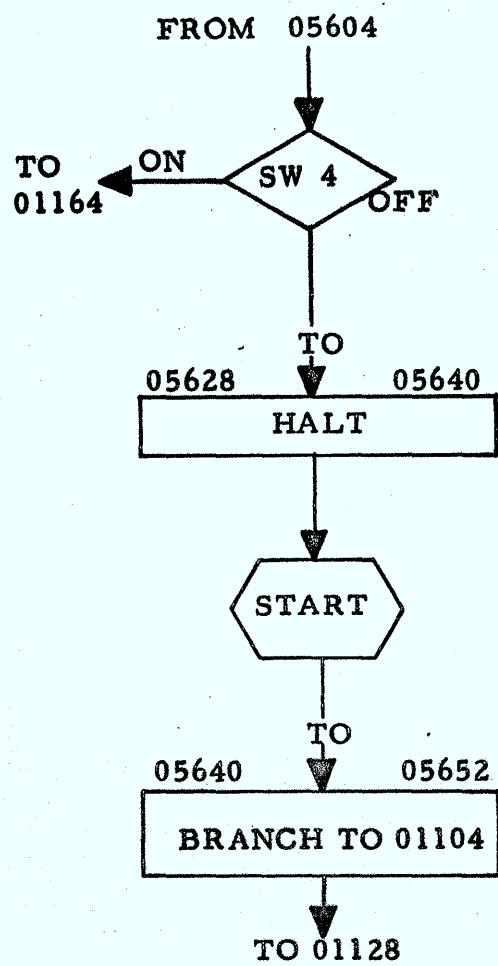
PN 2125574
EC 404568



A "HANG-UP" IN 1 TIME WITH A 47 OP CODE AT INSTRUCTION 05076 INDICATES IA IS NOT BEING INHIBITED.

A "HANG-UP" IN 1 TIME WITH A 41 OP CODE AT INSTRUCTION ON 05148 INDICATES IA IS NOT BEING INHIBITED.

A "HANG-UP" IN 1 TIME WITH A 34 OP CODE AT INSTRUCTION 05232 INDICATES IA IS NOT BEING INHIBITED.



A "HANG-UP" INDICATES IA IS NOT BEING INHIBITED.

TO PRODUCE A NEW TAPE:

1. LOAD MASTER TAPE INTO MEMORY STARTING 14444
2. DUMP NUMERIC TO PAPER TAPE PUNCH.
3. WHEN PROGRAM HALTS, NEW TAPE HAS BEEN GENERATED.

RESET

INSERT

KEY IN 3 6 1 4 4 4 4 0 0 3 0 0
3 5 1 4 4 4 4 4 0 0 2 0 0
4 8

RELEASE

START

PN 2125574
EC 404568

1620 DIAGNOSTICS

CU03 INDIRECT ADDRESSING TEST

MEM LOC	DESCRIPTION			
	00 01	PPPPP 23456	QQQQQ 78901	TYP
096		000	00000	MT
108	00	00102	03040	MT
120	00	20406	08000	MT
132	30	60902	10040	MT
144	80	21610	05001	MT
156	51	02006	02181	MT
168	42	00704	11282	MT
180	00	80614	22300	MT
192	90	81726	30000	MT
204	00	00005	06070	MT
216	80	90012	14161	MT
228	81	51811	24272	MT
240	02	42822	36352	MT
252	03	53045	40363	MT
264	24	84455	32494	MT
276	65	36048	46546	MT
288	27	54453	62718	MT
300	01	23456	78912	AT
312	34	56789	02345	AT
324	67	89013	45678	AT
336	90	12456	78901	AT
348	23	56789	01234	AT
360	67	89012	34578	AT
372	90	12345	68901	AT
384	23	45679	01234	AT
396	56	78 #		AT
408				X
420				X
432				X
444				X
456				X
468				X
480				X
492				X
504				X

ROUTINE 001

TYPE OUT SENSE SWITCH SETTING
AND NAME OF TEST

516	62	66 7	1 56	X	SW 1 0
528	55	0#6	266	X	N# SW
540	71	564	646	X	1 OFF
552	0#	6266	72	X	#SW 2
564	56	55 0	#6266	X	ON #SW
576		72 5	64646	X	2 OFF
588		0# 626	6 73	X	#SW 3
600		5655	0#62	X	ON #S
612	66	73	5646	X	W 3 OF
624	46	0#6	266	X	F #SW
636	74	565	5 0#	X	4 ON #
648	62	66 7	4 56	X	SW 4 0
660	46	46 0	76245	X	FF #SE
672	63	626	662	X	T SWS
684	46	5659	4955	X	FOR IN
696	44	49594	54363	X	DIRECT
708		41444	45945	X	ADDRE
720	62	62495	547	X	SSING
732	63	45626	303	X	TEST.
744	63	48455	5 62	X	THEN S
756	63	41596	303	X	TART
768	0#	62634	15963		# START
780	00	59566	46349		ROUTI
792	55	45620	30045		NES. E
804	63	56620	04656		TOS FO
816	53	53566	6030#		LLOW.#
828	46	00852	00100	BI	CHECK FOR SW1 ON
840	47	00876	00100	BNI	CHECK FOR SW1 OFF
852	39	00517	00100	WA	SW 1 ON
864	49	00888		B	

876	39	00535	00100	WA	SW 1 OFF
888	46	00912	00200	BI	CHECK FOR SW 2 ON
900	47	00936	00200	BNI	CHECK FOR SW 2 OFF
912	39	00555	00100	WA	SW 2 ON
924	49	00948		B	
936	39	00573	00100	WA	SW 2 OFF
948	46	00972	00300	BI	CHECK FOR SW 3 ON
960	47	00996	00300	BNI	CHECK FOR SW 3 OFF
972	39	00593	00100	WA	SW 3 ON
984	49	01008		B	
996	39	00611	00100	WA	SW 3 OFF
1008	46	01032	00400	BI	CHECK FOR 4 ON
1020	47	01056	00400	BNI	CHECK FOR SW 4 OFF
1032	39	00631	00100	WA	SW 4 ON
1044	49	01068		B	
1056	39	00649	00100	WA	SW 4 OFF
1068	39	00669	00100	WA	SET SWS FOR IA THEN START
1080	48			H	
1092	34		00102	K	CARRIAGE RETURN
1104	39	00771	00100	WA	ETOS FOLLOW
1116	34		00102	K	
1140	49	01224		B	
1152				X	
1164				X	
1176				X	
1188				X	

ROUTINE 002

CHECK FOR I/A Q. FIELD, EVEN POSITION,
ON ADD OPERATION

1200	01	210	077	X	CONSTANTS AND WORKING AREA
1212				X	WORKING AREA
1224	26	01207	01210	TF	SET AUGEND
1236	21	01207	01204	A	ADD WITH INDIRECT ADDRESS
1248	14	01207	154	C	CHECK FOR CORRECT ANSWER
1260	46	01368	01200	BI	CHECK E/Z FOR E/Z
1272	14	01207	287	C	CHECK FOR ERROR ANSWER
1284	46	01320	01200	BI	CHECK FOR E/Z FOR E/Z
1296	48	00002		H	
1308	49	01368		X	

ERROR ROUTINE

1320	46	01344	00100	BI
1332	39	01357	00100	WA
1344	47	01368	00300	BNI
1356	48	70707	2 0*	H
1368	46	01224	00200	BI
1380	49	01416		B

ROUTINE 003

CHECK FOR I/A Q FIELD, ODD POSITION,
ON ADD OPERATION

1392	0	1403	099	X	CONSTANTS AND WORKING AREA
1404				X	WORKING AREA
1416	26	01400	01403	TF	SET AUGEND
1428	21	01400	01397	A	ADD WITH INDIRECT ADDRESS
1440	14	01400	198	CM	CHECK FOR CORRECT ANSWER
1452	46	01560	01200	BI	CHECK E/Z FOR E/Z
1464	14	01400	502	CM	CHECK FOR ERROR ANSWER
1476	46	01512	01200	BI	CHECK E/Z FOR E/Z
1488	48	00003		H	
1500	49	01560	00100	BI	

ERROR ROUTINE

1512	46	01536	00100	BI
1524	39	01549	00100	WA
1536	47	01560	00300	BNI
1548	48	70707	3 0*	H
1560	46	01416	00200	BI
1572	49	01596		B

ROUTINE 004

CHECK FOR I/A P FIELD, EVEN POSITION
ON SUBTRACT OPERATION

1584	01	592	099	X	CONSTANTS AND WORKING AREA
1596	26	01592	01595	TF	SET MINUEND
1608	12	01588	066	SM	SUBTRACT WITH INDIRECT ADDRESS
1620	14	01592	033	CM	CHECK FOR CORRECT ANSWER
1632	46	01752	01200	BI	CHECK E/Z FOR E/Z
1644	14	01588	00526	CM	CHECK FOR ERROR ANSWER
1656	46	01692	01200	BI	CHECK E/Z FOR E/Z
1668	48	00004		H	
1680	49	01692		B	

ERROR ROUTINE

1692	16	01588	592	TFM	RESTORE CONSTANTS
1704	46	01728	00100	BI	
1716	39	01741	00100	WA	
1728	47	01752	00300	BNI	
1740	48	70707	4 0‡	H	
1752	46	01596	00200	BI	
1764	49	01788	0	B	

ROUTINE 005

CHECK FOR I/A P FIELD, ODD POSITION
ON SUBTRACT OPERATION

1776	17	83 0	099	X	CONSTANTS AND WORKING AREA
1788	26	01783	01786	TF	SET MINUEND
1800	12	01779	077	SM	SUBTRACT WITH INDIRECT ADDRESS
1812	14	01783	022	CM	CHECK FOR CORRECT ANSWER
1824	46	01944	01200	BI	CHECK E/Z FOR E/Z
1836	14	01779	706	CM	CHECK FOR ERROR ANSWER
1848	46	01884	01200	BI	CHECK E/Z FOR E/Z
1860	48	00005		H	
1872	49	01884		B	

ERROR ROUTINE

1884	16	01779	783	TFM	RESTORE CONSTANTS
1896	46	01920	00100	BI	
1908	39	01933	00100	WA	
1920	47	01944	00300	BNI	
1932	48	70707	5 0‡	H	
1944	46	01788	00200	BI	
1956	49	02004		B	

ROUTINE 006

CHECK FOR TWO I/A'S, Q FIELD, TWO EVEN
POSITION, ON A MULTIPLY OPERATION

1968	01	976	99666	X	CONSTANTS
1980	01	972		X	CONSTANTS
1992				X	WORKING AREA
2004	23	01979	01984	M	MULTIPLY WITH 2 INDIRECT ADDRESSES

2016	14	00099	65934	CM	CHECK FOR CORRECT ANSWER
2028	46	02136	01200	BI	CHECK E/Z FOR E/Z
2040	14	00099	47952	CM	CHECK FOR ERROR ANSWER
2052	46	02088	01200	BI	CHECK E/Z FOR E/Z
2064	48	00006		H	
2076	49	02136		B	

ERROR ROUTINE

2088	46	02112	00100	BI	
2100	39	02125	00100	WA	
2112	47	02136	00300	BNI	
2124	48	70707	6 0‡	H	
2136	46	02004	00200	BI	
2148	49	02196		B	

ROUTINE 007

CHECK FOR TWO I/A'S, Q FIELD, TWO ODD POSITION, ON A MULTIPLY OPERATION

2160	0	21677	7 888	X	CONSTANTS
2172	0	2165		X	CONSTANTS
2184				X	WORKING AREA
2196	23	02171	02177	M	MULTIPLY WITH 2 INDIRECT ADDRESSES
2208	14	00099	68376	CM	CHECK FOR CORRECT ANSWER
2220	46	02328	01200	BI	CHECK E/Z FOR E/Z
2232	14	00099	57720	CM	CHECK FOR ERROR ANSWER
2244	46	02280	01200	BI	CHECK E/Z FOR E/Z
2256	48	00007		H	
2268	49	02328		B	

ERROR ROUTINE

2280	46	02304	00100	BI	
2292	39	02317	00100	WA	
2304	47	02328	00300	BNI	
2316	48	70707	7 0‡	H	
2328	46	02196	00200	BI	
2340	49	02388		B	

ROUTINE 008

CHECK FOR TWO I/A'S, Q FIELD, ODD-EVEN
POSITION, ON A MULTIPLY OPERATION

2352	0	2360	33444	X	CONSTANTS
2364	02	357		X	CONSTANTS
2376				X	WORKING AREA
2388	23	02363	02368	M	MULTIPLY WITH 2 INDIRECT ADDRESSES
2400	14	00099	14652	CM	CHECK FOR CORRECT ANSWER
2412	46	02520	01200	BI	CHECK E/Z FOR E/Z
2424	14	00099	25308	CM	CHECK FOR ERROR ANSWER
2436	46	02472	01200	BI	CHECK E/Z FOR E/Z
2448	48	00008		H	
2460	49	02520		B	

ERROR ROUTINE

2472	46	02496	00100	BI	
2484	39	02509	00100	WA	
2496	47	02520	00300	BNI	
2508	48	70707	8 0‡	H	
2520	46	02388	00200	BI	
2532	49	02580		B	

ROUTINE 009

CHECK FOR TWO I/A'S, P FIELD, EVEN POSITION
ON A TRANSMIT FIELD OPERATION

2544	02	566‡	70248	X	CONSTANTS
2556	02	548‡0		X	CONSTANTS
2568				X	WORKING AREA
2580	26	02560	02555	TF	TRANSMIT FIELD WITH 2 INDIRECT ADDRESSES
2592	14	02566	70248	CM	CHECK FOR CORRECT ANSWER
2604	47	02652	01200	BNI	CHECK E/Z FOR E/Z
2616	16	02566	00000	TFM	RESTORE WORKING AREA TO ZERO
2628	16	02560	02548	TFM	RESTORE WORKING AREA
2640	49	02748		B	

ERROR ROUTINE

2652	46	02700	00100	BI	
2664	39	02713	00100	WA	
2676	38	02556	00100	WN	
2688	38	02544	00100	WN	
2700	47	02724	00300	BNI	

2712	48	70707	9 0‡	H
2724	16	02560	02548	TFM
2736	16	02548	02566	TFM
2748	46	02580	00200	BI
2760	49	02808		B

ROUTINE 010

CHECK FOR TWO I/A'S, P FIELD ODD POSITION,
ON A TRANSMIT FIELD OPERATION

2772	0	2795‡	82047	X	CONSTANTS
2784	0	2777‡	0	X	CONSTANTS
2796				X	WORKING AREA
2808	26	02789	02783	TF	TRANSMIT FIELD WITH 2 INDIRECT ADDRESSES
2820	14	02795	82047	CM	CHECK FOR CORRECT ANSWER
2832	47	02880	01200	BNI	CHECK E/Z FOR E/Z
2844	16	02795	00000	TFM	RESTORE WORKING AREA TO ZERO
2856	49	02976		B	
2868				X	

ERROR ROUTINE

2880	46	02928	00100	BI
2892	39	02941	00100	WA
2904	38	02785	00100	WN
2916	38	02773	00100	WN
2928	47	02952	00300	BNI
2940	48	70717	0 0‡	H
2952	16	02789	02777	TFM
2964	16	02777	02795	TFM
2976	46	02808	00200	BI
2988	49	03036		B

ROUTINE 011

CHECK FOR NO I/A ON ADD IMMEDIATE,
Q FIELD

3000	88	88‡	03011	X	CONSTANTS
3012			0222	X	CONSTANTS
3024				X	WORKING AREA
3036	16	03003	8888	TFM	SET AUGEND
3048	11	03003	03011	AM	CHECK FOR NO I/A ON ADD IMMEDIATE
3060	14	03003	5877	CM	CHECK FOR CORRECT ANSWER
3072	46	03192	01200	BI	CHECK E/Z FOR E/Z

3084	14	03003	9110	CM	CHECK FOR ERROR ANSWER
3096	46	03144	01200	BI	CHECK E/Z FOR E/Z
3108	38	03000	00100	WN	TYPE ANSWER
3120	48	00011		H	HALT
3132	41			NOP	

ERROR ROUTINE

3144	46	03168	00100	BI
3156	39	03181	00100	WA
3168	47	03192	00300	BNI
3180	48	70717	1 0*	H
3192	46	03036	00200	BI
3204	49	03240		B

ROUTINE 012

CHECK FOR NO I/A ON SUBTRACT IMMEDIATE, Q FIELD

3216	44	03222	X	CONSTANTS	
3228			X	WORKING AREA	
3240	16	03219	7777	TFM	SET MINUEND
3252	12	03219	03227	SM	CHECK FOR NO I/A SUBTRACT IMMEDIATE
3264	14	03219	8004	CM	CHECK FOR CORRECT ANSWER
3276	46	03396	01200	BI	CHECK E/Z FOR E/Z
3288	14	03219	7333	CM	CHECK FOR ERROR ANSWER
3300	46	03348	01200	BI	CHECK E/Z FOR E/Z
3312	38	03216	00100	WN	TYPE ANSWER
3324	48	00012		H	HALT
3336	41			NOP	

ERROR ROUTINE

3348	46	03372	00100	BI
3360	39	03385	00100	WA
3372	47	03396	00300	BNI
3384	48	70717	2 0*	H
3396	46	03240	00200	BI
3408	49	03444		B

ROUTINE 013

CHECK FOR NO I/A ON MULTIPLY IMMEDIATE, Q FIELD

3420	11	122	03424	X	CONSTANTS
3432				X	WORKING AREA

3444	13	03422	03431	MM	MULTIPLY IMMEDIATE
3456	14	00099	3441	CM	CHECK FOR CORRECT ANSWER
3468	46	03588	01200	BI	CHECK E/Z FOR E/Z
3480	14	00099	2442	CM	CHECK FOR ERROR ANSWER
3492	46	03540	01200	BI	CHECK E/Z FOR E/Z
3504	48	00013		H	HALT
3516	49	03540		B	
3528				X	

ERROR ROUTINE

3540	46	03564	00100	BI	
3552	39	03577	00100	WA	
3564	47	03528	00300	BNI	
3576	48	70717	3 07	H	
3588	46	03444	00200	BI	
3600	49	03672		B	
3612				X	
3624				X	
3636				X	
3648				X	

ROUTINE 014

**CHECK NO I/A ON COMPATE IMMEDIATE,
Q FIELD**

3660	03	671	55555	X	CONSTANTS AND WORKING AREA
3672	14	03664	03671	CM	CHECK FOR NO I/A ON COMP IMMED Q FIELD
3684	47	03732	01200	BNI	CHECK E/Z FOR E/Z
3696	49	03780		B	
3708				X	
3720				X	

ERROR ROUTINE

3732	46	03756	00100	BI	
3744	39	03769	00100	WA	
3756	47	03780	00300	BNI	
3768	48	70717	4 07	H	
3780	46	03672	00200	BI	
3792	49	03816		B	

ROUTINE 015

CHECK FOR NO I/A ON TRANSMIT DIGIT
IMMEDIATE, Q FIELD

3804	65	9 6	03806	X	WORKING AREA
3816	15	03809	03815	TDM	CHECK FOR NO IA ON TDM Q
3828	24	03809	03805	C	CHECK CORRECT DIGIT TRANSFERRED
3840	47	03888	01200	BNI	CHECK E/Z FOR E/Z
3852	15	03809	0	TDM	RESTORE WORKING AREA
3864	49	03936		B	
3876				X	

ERROR ROUTINE

3888	46	03912	00100	BI	
3900	39	03925	00100	WA	
3912	47	03936	00300	BNI	
3924	48	70717	5 0#	H	
3936	46	03816	00200	BI	
3948	49	03984		B	

ROUTINE 016

CHECK FOR NO I/A ON TRANSMIT FIELD
IMMEDIATE, Q FIELD

3960		1287	X	WORKING AREA	
3972		03971	X	CONSTANTS	
3984	16	03966	03971	TFM	CHECK FOR NO IA ON TFM Q FIELD
3996	24	03966	03983	C	CHECK CORRECT DATA TRANSFERRED
4008	47	04044	01200	BNI	CHECK E/Z FOR E/Z
4020	16	03966	00000	TFM	RESTORE WORKING AREA
4032	49	04092		X	

ERROR ROUTINE

4044	46	04068	00100	BI	
4056	39	04081	00100	WA	
4068	47	04092	00300	BNI	
4080	48	70717	6 0#	H	
4092	46	03984	00200	BI	
4104	49	04272		B	

PN 2125574
EC 404568

ROUTINE 017

CHECK FOR NO I/A ON WRITE
ALPHABETIC, Q FIELD

4116	63	0#	X	T
4128			X	
4140			X	
4152			X	
4164			X	
4176			X	
4188			X	
4200			X	
4212			X	
4224			X	
4236			X	
4248			X	
4260			X	
4272	16	10100	04295	TFM SET DATA IN IA FIELD
4284	39	04117	10100	WA CHECK FOR NO IA IN Q ON WA
4296	49	04356		B
4308				X
4320				X

ROUTINE 018

CHECK FOR I/A ON TRANSMIT
RECORD, Q FIELD

4332			04351	X	CONSTANTS AND WORKING AREA
4344	28	#	968#	X	CONSTANTS AND WORKING AREA
4356	31	04332	04343	TR	CHECK FOR IA IN Q FIELD ON TR
4368	14	04334	968	CM	CHECK FOR CORRECT TRANSFER
4380	47	04428	01200	BNI	CHECK E/Z FOR E/Z
4392	49	04476		B	
4404				X	
4416				X	

ERROR ROUTINE

4428	46	04452	00100	BI
4440	39	04465	00100	WA
4452	47	04476	00300	BNI
4464	48	70717	8 0#	H
4476	46	04356	00200	BI
4488	49	04512		B

ROUTINE 019

CHECK FOR NO I/A ON BRANCH AND
TRANSMIT IMMEDIATE, Q FIELD

4500		98765	04506	X	CONSTANTS AND WORKING AREA
4512	17	04644	04511	BTM	CHECK NO IA ON BTM
4524	48	04643	00000	X	OP CODE 48 OR 16 IF HALT 519 NOT PERFORMED
4536	16	04525	48	TFM	SET P CODE 48 IN
4548	49	04608		B	

ERROR ROUTINE

4560	46	04584	00100	BI
4572	39	04597	00100	WA
4584	47	04608	00300	BNI
4596	48	70717	9 0#	H
4608	46	04512	00200	BI
4620	49	04740		B

ROUTINE 519

THIS IS ROUTINE BRANCHED TO IN 019
CHECKS FOR NO I/A ON BRANCH BACK

4632			X	WORKING AREA	
4644	14	04643	04511	CM	CHECK TF OF ROUTINE 019 CORRECT
4656	47	04560	01200	BNI	CHECK E/Z TRIG FOR E/Z
4668	16	04525	16	TFM	SET OP CODE 16 IN 04524
4680	42	04691	04686	BB	IF HANGS UP IN THIS STEP IA ON BB
4692			X		
4704			X		

PN 2125574
EC 404568

ROUTINE 020

CHECK FOR NO I/A ON BRANCH
INDICATE, Q FIELD

4716		000	X	WORKING AREA
4728			X	WORKING AREA
4740	11	04727	01	AM SET H/P TRIG H/P
4752	16	11104	04775	TFM SET UP IA AT 11104
4764	46	04788	11104	BI CHECK NO IA ON BI Q FIELD
4776	49	04824		B ENTER ERROR ROUTINE
4788	16	04727	000	TFM CLEAR ADD FIELD
4800	49	04872		B
4812			X	

ERROR ROUTINE

4824	46	04848	00100	BI
4836	39	04861	00100	WA
4848	47	04872	00300	BNI
4860	48	70727	0 0	H
4872	46	04740	00200	BI
4884	49	04908		B

ROUTINE 021

CHECK FOR NO I/A ON BRANCH NO
INDICATE, Q FIELD

4896		00	X	WORKING AREA
4908	11	04907	22	AM SET H/P TRIG H/P
4920	16	11109	04943	TFM SET UP IA AT 11109
4932	47	04992	11109	BNI CHECK NO IA ON BNI
4944	16	04907	000	TFM CLEAR ADD FIELD
4956	49	05040		B
4968			X	
4980			X	

ERROR ROUTINE

4992	46	05016	00100	BI
5004	39	05029	00100	WA
5016	47	05040	00300	BNI
5028	48	70727	1 0#	H
5040	46	04908	00200	BI
5052	49	05064		B

ROUTINE 022

CHECK FOR NO I/A ON BRANCH,
Q FIELD

5064	16	11114	05087	TFM	SET UP IA AT 11114
5076	49	05112	11114	B	CHECK NO IA ON BRANCH
5088				X	
5100				X	
5112	46	05064	00200	BI	
5124	49	05148		B	

ROUTINE 023

CHECK FOR NO I/A ON NO OPOSITION
P AND Q FIELDS

5136				X	
5148	41	05159	05154	NOP	CHECK FOR NO IA ON NOP
5160	49	05184		B	
5172				X	
5184	46	05148	00200	BI	
5196	49	05220		B	

ROUTINE 024

CHECK FOR NO I/A ON CONTROL,
P FIELD

5208				X	
5220	16	10101	05238	TFM	SET 05238 AT 10101
5232	34	10101	10101	K	CHECK NO IA ON CONTROL P FIELD
5244	49	05280		B	
5256				X	

ROUTINE 025

TIMES 100 ROUTINE

5268			000	X	WORKING AREA
5280	46	05292	01400	BI	TURN OFF OVERFLOW
5292	11	05279	10	AM	ADD ONE
5304	47	01224	01400	BNI	CHECK FOR OVERFLOW
5316	49	05592		B	

ROUTINE 026

TEST COMPLETED ROUTINE

5328				X	
5340				X	
5352	63	45626	3 59	X	TEST R
5364	56	64634	95545	X	OUTINE
5376	62	435	65457	X	S COMP
5388	53	45634	54403	X	LETED
5400		4946	6266	X	IF SW
5412	71	564	64623	X	I OFE,
5424		5556	5956	X	NO RO
5436	64	63495	545	X	UTINE
5448	55	5662	6368	X	NOS TY
5460	57	4544	5664	X	PED OU
5472	63	23 4	15544	X	T, AND
5484		5556	4841	X	NO HA
5496	55	47206	45762	X	NG-UPS
5508	23	544	14348	X	, MACH
5520	49	5545	5745	X	INE PE
5532	59	46565	95445	X	RFORME
5544	44	634	56263	X	TEST
5556	62	575	95657	X	S PROP
5568	45	59536	803	X	ERLY.
5580	#			X	#
5592	34		00102	K	CARRIAGE RETURN
5604	39	05353	00100	WA	TYPE TEST COMPLETED
5616	46	01092	00400	BI	
5628	48	05634	05639	H	
5640	49	01104	E	B	

CS01– Check Stops Diagnostic


```
=====  
//  
// CS01 - Check Stops Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: not used  
// PS2: not used  
// PS3: not used  
// PS4: not used  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - STOP  
//  
// Start addresses:  
//  
// 00402 - Full test  
//  
// Directions:  
//  
// 1. Load CS01 diagnostic  
// 2. Press START  
// 3. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/C-8-2  
// 4. Press RESET  
// 5. Press START  
// 6. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C-8-2/C  
// 7. Press RESET  
// 8. Press START  
// 9. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C-8-2/C/C  
// 10. Press RESET  
// 11. Press START  
// 12. Verify CHECK STOP & MAR CHK are on and MAR = C/C-8-2/C/C/C  
// 13. Press RESET  
// 14. Press START  
// 15. Verify CHECK STOP & MAR CHK are on and MAR = C-8-2/C/C/C/C  
// 16. Press RESET  
// 17. Press START  
// 18. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/8-2-1  
// 19. Press RESET  
// 20. Press START  
// 21. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/8-2-1/C  
// 22. Press RESET  
// 23. Press START  
// 24. Verify CHECK STOP & MAR CHK are on and MAR = C/C/8-2-1/C/C  
// 25. Press RESET  
// 26. Press START  
// 27. Verify CHECK STOP & MAR CHK are on and MAR = C/8-2-1/C/C/C  
// 28. Press RESET  
// 29. Press START  
// 30. Verify CHECK STOP & MAR CHK are on and MAR = 8-2-1/C/C/C/C  
// 31. Press RESET  
// 32. Press START  
// 33. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/C-8-4  
// 34. Press RESET  
// 35. Press START  
// 36. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C-8-4/C
```

```
// 37. Press RESET
// 38. Press START
// 39. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C-8-4/C/C
// 40. Press RESET
// 41. Press START
// 42. Verify CHECK STOP & MAR CHK are on and MAR = C/C-8-4/C/C/C
// 43. Press RESET
// 44. Press START
// 45. Verify CHECK STOP & MAR CHK are on and MAR = C-8-4/C/C/C/C
// 46. Press RESET
// 47. Press START
// 48. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/8-4-1
// 49. Press RESET
// 50. Press START
// 51. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/8-4-1/C
// 52. Press RESET
// 53. Press START
// 54. Verify CHECK STOP & MAR CHK are on and MAR = C/C/8-4-1/C/C
// 55. Press RESET
// 56. Press START
// 57. Verify CHECK STOP & MAR CHK are on and MAR = C/8-4-1/C/C/C
// 58. Press RESET
// 59. Press START
// 60. Verify CHECK STOP & MAR CHK are on and MAR = 8-4-1/C/C/C/C
// 61. Press RESET
// 62. Press START
// 63. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/8-4-2
// 64. Press RESET
// 65. Press START
// 66. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/8-4-2/C
// 67. Press RESET
// 68. Press START
// 69. Verify CHECK STOP & MAR CHK are on and MAR = C/C/8-4-2/C/C
// 70. Press RESET
// 71. Press START
// 72. Verify CHECK STOP & MAR CHK are on and MAR = C/8-4-2/C/C/C
// 73. Press RESET
// 74. Press START
// 75. Verify CHECK STOP & MAR CHK are on and MAR = 8-4-2/C/C/C/C
// 76. Press RESET
// 77. Press START
// 78. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C/C-8-4-2-1
// 79. Press RESET
// 80. Press START
// 81. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C/C-8-4-2-1/C
// 82. Press RESET
// 83. Press START
// 84. Verify CHECK STOP & MAR CHK are on and MAR = C/C/C-8-4-2-1/C/C
// 85. Press RESET
// 86. Press START
// 87. Verify CHECK STOP & MAR CHK are on and MAR = C/C-8-4-2-1/C/C/C
// 88. Press RESET
// 89. Press START
// 90. Verify CHECK STOP & MAR CHK are on and MAR = C-8-4-2-1/C/C/C/C
// 91. Press RESET
// 92. Press START
// 93. Verify CHECK STOP is on and OPERATION REGISTER = 7/7
// 94. Press RESET
// 95. Press START
// 96. Verify CHECK STOP is on and OPERATION REGISTER = C/C-8-4-2-1
```

```
// 97. Press RESET
// 98. Press START
// 99. Verify CHECK STOP is on and OPERATION REGISTER = C-8-4-2-1/C
// 100. Press RESET
// 101. Press START
// 102. Verify CHECK STOP & MAR CHK are on and MAR = C/8/4/2/1
// 103. Press RESET
// 104. Press INSERT
// 105. Press RELEASE
// 106. Press START
// 107. Verify CHECK STOP & MAR CHK are on and MAR = C-4-2/C/C/C/C
// 108. Press RESET
// 109. Press START
// 110. Verify CHECK STOP & MAR CHK are on and MAR = C-8-1/C-8-1/C-8-1/C-8-1/C-8-1
// 111. Press RESET
// 112. Press START
// 113. Type A
// 114. Verify CHECK STOP & MAR CHK are on and MAR = 1/C/C/C/C
// 115. Press RESET
// 116. Press START
// 117. Verify CHECK STOP & MAR CHK are on and MAR = 1/C/C/C/C
//=====
=====
```


Sample Output - CS01

CHECK STOP TEST (CSØ1)

MAR CHECKS - NON-NUMERIC DIGITS IN EVERY POSITION

- DIGIT 8-2 (5)
- DIGIT 8-2-1 (5)
- DIGIT 8-4 (5)
- DIGIT 8-4-1 (5)
- DIGIT 8-4-2 (5)
- DIGIT 8-4-2-1 (5)

OPCODE CHECKS - BAD INSTRUCTIONS

- UNDEFINED OPCODE (1)
- NON-NUMERIC OPCODES (2)
- UNALIGNED INSTRUCTION (1)

MEMORY CHECKS - BAD ADDRESSES

- OUT-OF-RANGE (2)

IO CHECK - UNALIGNED READ/WRITE

- READ ALPHAMERIC (1)
A
- WRITE ALPHAMERIC (1)

END OF TESTS

// CS01 - Check Stops diagnostic

00000: 04 09 00 00 04 00 02 00 00 00 00 00 00 04 08 00 00 00 00 00 00 00 // "I - H " 00020: 00 00 00 00 04 09 00 00 04 00 02 00 00 00 00 00 00 00 00 00 // " I - " 00100: 00 00 00 00 00 00 00 00 00 00 00 00 01 00 02 00 03 00 04 00 // " +-" 00120: 00 00 02 00 04 00 06 00 08 00 00 00 03 00 06 00 09 00 02 01 // " - /" 00140: 00 00 04 00 08 00 02 01 06 01 00 00 05 00 00 01 05 01 00 02 // " / -J" 00160: 00 00 06 00 02 01 08 01 04 02 00 00 07 00 04 01 01 02 08 02 // " /B 0A" 00180: 00 00 08 00 06 01 04 02 02 03 00 00 09 00 08 01 07 02 06 03 // " B, 2T" 00200: 00 00 00 00 00 00 00 00 00 00 05 00 06 00 07 00 08 00 09 00 // " -0" 00220: 00 01 02 01 04 01 06 01 08 01 05 01 08 01 01 02 04 02 07 02 // "/A JB2" 00240: 00 02 04 02 08 02 02 03 06 03 05 02 00 03 05 03 00 04 05 04 // "B, T K.L)M" 00260: 00 03 06 03 02 04 08 04 04 05 05 03 02 04 09 04 06 05 03 06 // ".T(E L(V" 00280: 00 04 08 04 06 05 04 06 02 07 05 04 04 05 03 06 02 07 01 08 // ")VF ME" 00300: 00 01 02 03 04 05 06 07 08 09 01 02 03 04 05 06 07 08 09 10 // ",EX @O8" 00320: 02 03 04 05 06 07 08 09 10 11 03 04 05 06 07 08 09 10 11 12 // ",EX @O8" 00340: 04 05 06 07 08 09 10 11 12 13 05 06 07 08 09 10 11 12 13 14 // "EX, O8@" 00360: 06 07 08 09 10 11 12 13 14 15 07 08 09 10 11 12 13 14 15 16 // "X, E 8@O" 00380: 08 09 10 11 12 13 14 15 16 17 09 10 11 12 13 14 15 16 17 18 // ",EX @O8" 00400: 0A 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 00 01 04 04 // "@ D" 00420: 07 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 04 // "0 @ @" 00440: 00 00 00 00 00 00 01 00 02 03 09 00 01 04 09 03 00 00 01 // " I" 00460: 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 00 01 05 09 // " @ R" 00480: 03 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 03 // " @ = " 00500: 00 00 00 0A 00 00 00 00 00 00 03 03 00 00 00 0A 00 00 00 00 // " = # " 00520: 00 00 03 03 00 00 0A 00 00 00 00 00 00 00 00 03 03 00 0A 00 00 // " = =# " 00540: 00 00 00 00 00 03 03 0A 00 00 00 00 00 00 00 00 00 00 03 09 // " = " 00560: 00 01 06 02 09 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 // "S @ " 00580: 00 02 03 03 00 00 00 00 0B 00 00 00 00 00 03 03 00 00 00 0B // "= =#" 00600: 00 00 00 00 00 03 03 00 00 0B 00 00 00 00 00 00 00 00 03 03 // " = = " 00620: 00 0B 00 00 00 00 00 00 00 03 03 0B 00 00 00 00 00 00 00 00 // "# = " 00640: 00 00 03 09 00 01 06 06 09 00 00 01 00 00 03 04 00 00 00 00 // " W @ " 00660: 00 00 00 01 00 02 03 03 00 00 00 00 0C 00 00 00 00 00 00 03 03 // " = = " 00680: 00 00 00 0C 00 00 00 00 00 00 03 03 00 00 0C 00 00 00 00 00 // " = " 00700: 00 00 03 03 00 0C 00 00 00 00 00 00 00 00 00 03 03 0C 00 00 00 // " = = " 00720: 00 00 00 00 00 03 09 00 01 07 00 05 00 00 01 00 00 03 04 // " 0- @" 00740: 00 00 00 00 00 01 00 02 03 03 00 00 00 00 00 0D 00 00 00 00 // " = " 00760: 00 00 03 03 00 00 00 0D 00 00 00 00 00 00 03 03 00 00 0D 00 // " = = " 00780: 00 00 00 00 00 03 03 00 0D 00 00 00 00 00 00 00 00 00 03 03 // " = = " 00800: 0D 00 00 00 00 00 00 00 00 03 09 00 01 07 04 05 00 00 01 // " = 4-" 00820: 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 03 00 00 00 00 // " @ = " 00840: 0E 00 00 00 00 03 03 00 00 00 0E 00 00 00 00 00 00 00 03 03 // " = = " 00860: 00 00 0E 00 00 00 00 00 00 03 03 00 0E 00 00 00 00 00 00 00 // " = = " 00880: 00 00 03 03 0E 00 00 00 00 00 00 00 00 00 03 09 00 01 07 08 // " = 8" 00900: 05 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 03 // "- @ = " 00920: 00 00 00 0F 00 00 00 00 00 03 03 00 00 00 0F 00 00 00 00 00 // " = # " 00940: 00 00 03 03 00 00 0F 00 00 00 00 00 00 00 03 03 00 0F 00 00 // " = =# " 00960: 00 00 00 00 00 03 03 0F 00 00 00 00 00 00 00 00 00 00 03 09 // " = " 00980: 00 01 08 02 09 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 // " @ " 01000: 00 02 03 09 00 01 08 09 05 00 00 01 00 00 03 04 00 00 00 00 // "- @ " 01020: 00 00 00 01 00 02 07 07 00 00 00 00 00 00 00 00 00 00 03 09 // " 7 " 01040: 00 01 09 04 05 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 // "- @ " 01060: 00 02 00 0F 00 00 00 00 00 00 00 00 00 00 0F 00 00 00 00 00 // "# " 01080: 00 00 00 00 00 03 09 00 02 00 00 01 00 00 01 00 00 03 04 // " + @" 01100: 00 00 00 00 00 01 00 02 02 06 00 00 00 01 01 00 02 03 // " +, " 01120: 07 05 04 09 00 08 04 02 01 00 00 00 00 00 02 06 00 00 00 01 // "5IB+ " 01140: 01 00 02 03 06 03 03 09 00 02 00 06 01 00 00 01 00 00 03 04 // "+, T + @" 01160: 00 00 00 00 00 01 00 02 03 09 00 02 01 02 01 00 00 01 // " + " 01180: 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 03 06 00 00 00 // " @ = " 01200: 00 00 00 00 00 03 03 09 09 09 09 00 00 00 00 00 03 09 // " = "

01220: 00 02 01 06 03 00 00 01 00 00 03 04 00 00 00 00 00 00 00 00 01 // " @ "

 01240: 00 02 03 09 00 02 02 02 07 00 00 01 00 00 03 04 00 00 00 00 00 // "0 @ "

 01260: 00 00 00 01 00 02 03 04 00 00 00 00 00 00 00 01 00 01 03 04 // " @ @ "

 01280: 00 00 00 00 00 00 01 00 01 03 04 00 00 00 00 00 00 00 00 01 // " @ "

 01300: 00 01 03 04 00 00 00 00 00 00 00 01 00 01 03 07 01 00 00 00 // " @ + "

 01320: 00 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 // " @ "

 01340: 00 02 02 07 05 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 // "- @ "

 01360: 00 02 03 09 01 00 00 00 00 00 00 01 00 00 02 06 00 00 00 01 // "+ "

 01380: 01 00 02 03 06 03 03 04 00 00 00 00 00 00 00 01 00 02 03 09 // "+, T@ "

 01400: 00 02 03 02 07 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 // "0 @ "

 01420: 00 02 04 08 00 00 00 00 00 00 00 00 00 00 04 09 00 00 04 00 // "H I "

 01440: 02 00 00 00 00 14 03 04 08 04 05 04 03 05 02 00 00 06 02 // "- CH ECK S"

 01460: 06 03 05 06 05 07 00 00 06 03 04 05 06 02 06 03 00 00 02 04 // "TOP T EST ("

 01480: 04 03 06 02 07 00 07 01 00 04 00 0A 15 04 04 01 05 09 00 00 // "CS01) #MAR "

 01500: 04 03 04 08 04 05 04 03 05 02 06 02 00 00 02 00 00 05 05 // "CHECK S - N"

 01520: 05 06 05 05 02 00 05 05 06 04 05 04 04 05 05 09 04 09 04 03 // "ON-NU MERIC"

 01540: 00 00 04 04 04 09 04 07 04 09 06 03 06 02 00 00 04 09 05 05 // " DIGI TS IN"

 01560: 00 00 04 05 06 05 04 05 05 09 06 08 00 00 05 07 05 06 06 02 // " EVER Y POS"

 01580: 04 09 06 03 04 09 05 06 05 05 00 0A 10 00 00 00 02 00 00 00 // "ITION # - "

 01600: 04 04 04 09 04 07 04 09 06 03 00 00 07 08 02 00 07 02 00 00 // "DIGIT 8-2 "

 01620: 02 04 07 05 00 04 00 0A 10 00 00 00 02 00 00 00 04 04 04 09 // "(5) # - DI"

 01640: 04 07 04 09 06 03 00 00 07 08 02 00 07 02 02 00 07 01 00 00 // "GIT 8 -2-1 "

 01660: 02 04 07 05 00 04 00 0A 10 00 00 00 02 00 00 00 04 04 04 09 // "(5) # - DI"

 01680: 04 07 04 09 06 03 00 00 07 08 02 00 07 04 00 00 02 04 07 05 // "GIT 8 -4 (5"

 01700: 00 04 00 0A 10 00 00 00 02 00 00 00 04 04 04 09 04 07 04 09 // ") # - DIGI"

 01720: 06 03 00 00 07 08 02 00 07 04 02 00 07 01 00 00 02 04 07 05 // "T 8-4 -1 (5"

 01740: 00 04 00 0A 10 00 00 00 02 00 00 00 04 04 04 09 04 07 04 09 // ") # - DIGI"

 01760: 06 03 00 00 07 08 02 00 07 04 02 00 07 02 00 00 02 04 07 05 // "T 8-4 -2 (5"

 01780: 00 04 00 0A 10 00 00 00 02 00 00 00 04 04 04 09 04 07 04 09 // ") # - DIGI"

 01800: 06 03 00 00 07 08 02 00 07 04 02 00 07 02 02 00 07 01 00 00 // "T 8-4 -2-1 "

 01820: 02 04 07 05 00 04 00 0A 15 06 05 07 04 03 05 06 04 04 04 05 // "(5) #O PCODE"

 01840: 00 00 04 03 04 08 04 05 04 03 05 02 06 02 00 00 02 00 00 00 // " CHEC KS - "

 01860: 04 02 04 01 04 04 00 00 04 09 05 05 06 02 06 03 05 09 06 04 // "BAD I NSTRU"

 01880: 04 03 06 03 04 09 05 06 05 05 06 02 00 0A 10 00 00 00 02 00 // "CTION S# - "

 01900: 00 00 06 04 05 05 04 04 04 05 04 06 04 09 05 05 04 05 04 04 // " UNDE FINED"

 01920: 00 00 05 06 05 07 04 03 05 06 04 04 04 05 00 00 02 04 07 01 // " OPCO DE (1"

 01940: 00 04 00 0A 10 00 00 00 02 00 00 00 05 05 05 06 05 05 02 00 // ") # - NON-"

 01960: 05 05 06 04 05 04 04 05 05 09 04 09 04 03 00 00 05 06 05 07 // "NUMER IC OP"

 01980: 04 03 05 06 04 04 04 05 06 02 00 00 02 04 07 02 00 04 00 0A // "CODES (2) #"

 02000: 10 00 00 00 02 00 00 00 06 04 05 05 04 01 05 03 04 09 04 07 // " - U NALIG"

 02020: 05 05 04 05 04 04 00 00 04 09 05 05 06 02 06 03 05 09 06 04 // "NED I NSTRU"

 02040: 04 03 06 03 04 09 05 06 05 05 00 00 02 04 07 01 00 04 00 0A // "CTION (1) #"

 02060: 15 04 04 05 05 04 05 06 05 09 06 08 00 00 04 03 04 08 04 05 // "MEMOR Y CHE"

 02080: 04 03 05 02 06 02 00 00 02 00 00 00 04 02 04 01 04 04 00 00 // "CKS - BAD"

 02100: 04 01 04 04 04 05 09 04 05 06 02 06 02 04 05 06 02 00 0A // "ADDRE SSES#"

 02120: 10 00 00 00 02 00 00 00 05 06 06 04 06 03 02 00 05 06 04 06 // " - O UT-OF"

 02140: 02 00 05 09 04 01 05 05 04 07 04 05 00 00 02 04 07 02 00 04 // "-RANG E (2)"

 02160: 00 0A 14 09 05 06 00 00 04 03 04 08 04 05 04 03 05 02 00 00 // "#IO C HECK"

 02180: 02 00 00 00 06 04 05 05 04 01 05 03 04 09 04 07 05 05 04 05 // "- UNA LIGNE"

 02200: 04 04 00 00 05 09 04 05 04 01 04 04 02 01 06 06 05 09 04 09 // "D REA D/WRI"

 02220: 06 03 04 05 00 0A 10 00 00 00 02 00 00 00 05 09 04 05 04 01 // "TE# - REA"

 02240: 04 04 00 00 04 01 05 03 05 07 04 08 04 01 05 04 04 05 05 09 // "D ALP HAMER"

 02260: 04 09 04 03 00 00 02 04 07 01 00 04 00 0A 10 00 00 00 02 00 // "IC (1) # - "

 02280: 00 00 06 06 05 09 04 09 06 03 04 05 00 00 04 01 05 03 05 07 // " WRIT E ALP"

 02300: 04 08 04 01 05 04 04 05 05 09 04 09 04 03 00 00 02 04 07 01 // "HAMER IC (1"

 02320: 00 04 00 00 00 0A 14 05 05 05 04 04 00 00 05 06 04 06 00 00 // ") #EN D OF "

 02340: 06 03 04 05 06 02 06 03 06 02 00 0A 14 09 00 00 04 00 02 00 // "TESTS #I - "

 02360: 00 00 00 00 14 09 00 01 01 03 04 00 00 00 00 00 00 00 00 00 // " I\$ "

// End

DX05– Core Storage 20K Diagnostic


```
//=====
//  
// DX05L - Core Storage L 20K Diagnostic  
//  
// Program Switch settings:  
//  
// PS1:   ON - Bypass error type out  
//         OFF - Type out routine number on error  
// PS2:   ON - Loop in routine  
//         OFF - Continue to next routine  
// PS3:   not used  
// PS4:   ON - Repeat test DX05L  
//         OFF - Run test DX05L once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY    - STOP  
// I/O        - STOP  
// O'FLOW     - STOP  
//  
// Start addresses:  
//  
// 05052 - Full test  
//  
// Directions:  
//  
// 1. Load DX05L diagnostic  
// 2. Press START  
//=====
```

```
//=====
//  
// DX05H - Core Storage H 20K Diagnostic  
//  
// Program Switch settings:  
//  
// PS1:   ON - Bypass error type out  
//         OFF - Type out routine number on error  
// PS2:   ON - Loop in routine  
//         OFF - Continue to next routine  
// PS3:   not used  
// PS4:   ON - Repeat test DX05H  
//         OFF - Run test DX05H once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY    - STOP  
// I/O        - STOP  
// O'FLOW     - STOP  
//  
// Start addresses:  
//  
// 00402 - Full test  
//  
// Directions:  
//  
// 1. Load DX05H diagnostic  
// 2. Press START  
//=====
```

Sample Output - DX05

DTXØ5L
PASS COMPLETE

DTXØ5H
PASS COMPLETE

NO. 2172342
SHEET 0
OF 19

DIAGNOSTIC TEST

TITLE CORE STORAGE TEST - L 20K - DTX054
MACH. TYPE 1620-1 BY HNJ APPR. DATE

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404980	5-7-64	1 - 19

E/C NO.	404980						
DATE	5-7-64						

Because this test destroys the contents of core, it is not DIPAL compatible.

PURPOSE OF TEST

This diagnostic tests the standard 20 K storage for marginal conditions by generating worst case half select noise patterns. It is divided into two parts which are loaded separately as if two different diagnostics.

5H - Tests all addresses ending in 50 through 99 (over 50 sense segment)

5L - Tests all addresses ending in 00 through 49 (under 50 sense segment)

The running time of either part is 36 seconds per pass.

SWITCH CONTROL

The Console Program Switches have the following control functions in this test:

Program Switch 1 ON - Bypass all error routines
 OFF - Type out errors

Program Switch 2 ON - Loop at test address
 OFF - Sequence through entire test

Program Switch 3 - Not Used

Program Switch 4 ON - Repeat Diagnostic
 OFF - Card Input: Load next program at end of pass

The PARITY Switch set in the PROGRAM position will allow the program to run without manual intervention but in the case where the error forms an illegal character it will be necessary to run with the switch in the STOP position in order to determine the failing bit.

The I/O and OVERFLOW Switches are set to the PROGRAM position.

TEST PROCEDURE

Card Input:

1. Clear Core Storage by inserting 31 00003 00002. Release and Start.
2. Load test deck from 1622 by pushing LOAD button or by inserting 36 00000 00300 R-S.

3. Both programs are on the same tape, however they must be loaded separately. The first program on the tape is X05H, the second is X05L.

To reproduce the Paper Tape, insert 3600000 00300, release, and press "SIE" button. DO NOT reposition the tape on the Reader after the program is read in, as the second program is in position to be read. After the program has been loaded, insert

LOC		
00000	38	00024
00012	35	00402
00024	49	00402
00036	36	00402
00048	36	10002
00060	49	00402

After MAR has counted to 04954, stop by pressing "SCE", reset, insert

LOC		
00000	35	19999
00012	35	10002
00024	35	19999
00036	48	

Release/start.

This is done to prevent writing 5000 zeros on tape.

When the first program, DTX05H, has been punched, clear core and insert 36 00000 00300 to read the second program, DTX05L. Release, and press "SIE" button.

After the program has been loaded, insert

LOC		
00000	38	00024
00012	35	05052
00024	49	05052
00036	36	05052
00048	36	15052
00060	49	05052

Release/start.

When MAR reaches 09604, stop by pressing "SCE" button, insert

LOC		
00000	35	19999
00012	35	15052
00024	35	19999
00036	48	

Release/start.

When MAR reaches 16779, stop by pressing "SCE" button, reset, and start.

4. 1620 will Halt with 00011 in MAR after deck is loaded. Push START to execute program.

The diagnostic was assembled by the SPS Assembly Program and uses a standard SPS Load routine. A detailed listing of the loader along with a brief introduction to the SPS Assembly Program may be found in the DTX02 Diagnostic write-up.

TEST DESCRIPTION

The pattern 7788 is written in the area to be tested and then each address in turn is complimented and then restored to its original value. Read out of the originally written character is with the half select noise in phase. Read out of the complimented character is with the half select noise out of phase. After all addresses are tested the pattern is reversed and the procedure is repeated so that every core is tested under both conditions.

In order to provide the best chance of successful test operation, the two parts of the test are confined to diagonally opposite corners of storage. No common X or Y drive line, or sense segments are used in the program area. There is no possibility of isolation in the address decode, matrix switch area.

The machine add and multiply tables are not used in this program as no arithmetic instructions have been used. The program is made up of information Transfer and Branch type instructions only.

Any reference to add tables in the listing or flow chart are to program tables, not the machine add table.

The pattern is first written throughout the tested area of storage with a Transmit Record instruction in blocks of fifty characters. If an error occurs during this phase of the program, the error routine must scan the fifty character field with a Transmit Digit instruction in order to find the specific address that failed.

The Compliment and Restore phase uses a Transmit Digit instruction, so the error address is always known. Because of the lengthy nature of this test, a rather large address modification loop is used in order to reduce the running time to a minimum. Ten addresses are completely checked each time through the loop. The pattern set up in the loop is 7788778877 (10 digits). It can be seen that it will be necessary to compliment this pattern each time through the loop in order to keep it progressing properly. In the flow chart, this is done in a clock labelled "Swap Test Digit Area".

ERROR HALTS

Explanation

00011	Program Loading Complete
04637	LH Pass complete and Program Switch 4 off. Waiting for card reader
09287	LL Pass complete and Program Switch 4 off. Waiting for card reader

ERROR PRINT OUT

In the pattern writing phase, an error print out may be one or two lines. An error in an even adr only will give a one line print out. An error in an odd address will give a two line print out. One for both the even and odd address.

Even address error:

X is error char XXXXX is error ADR (Even Address)

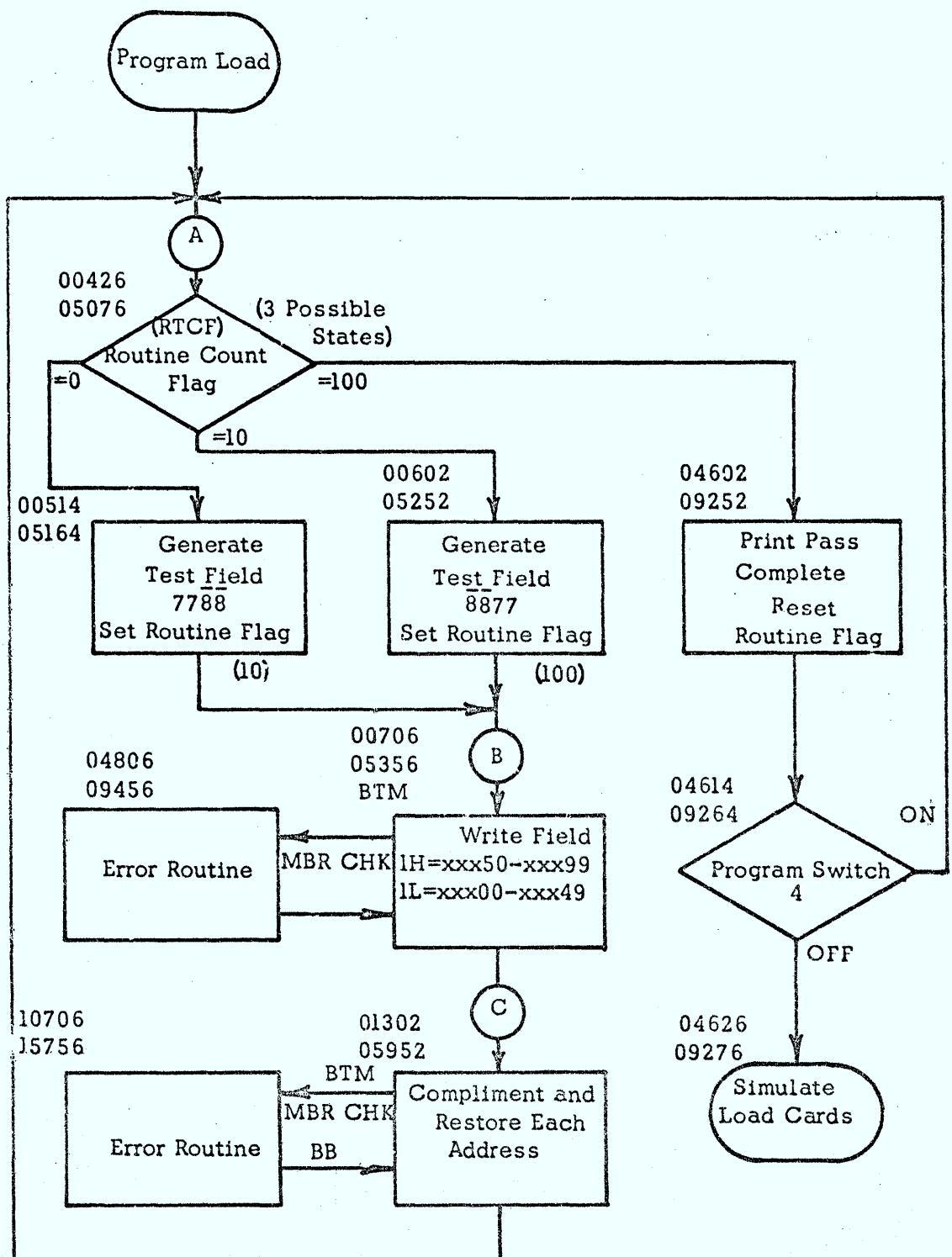
Odd address error:

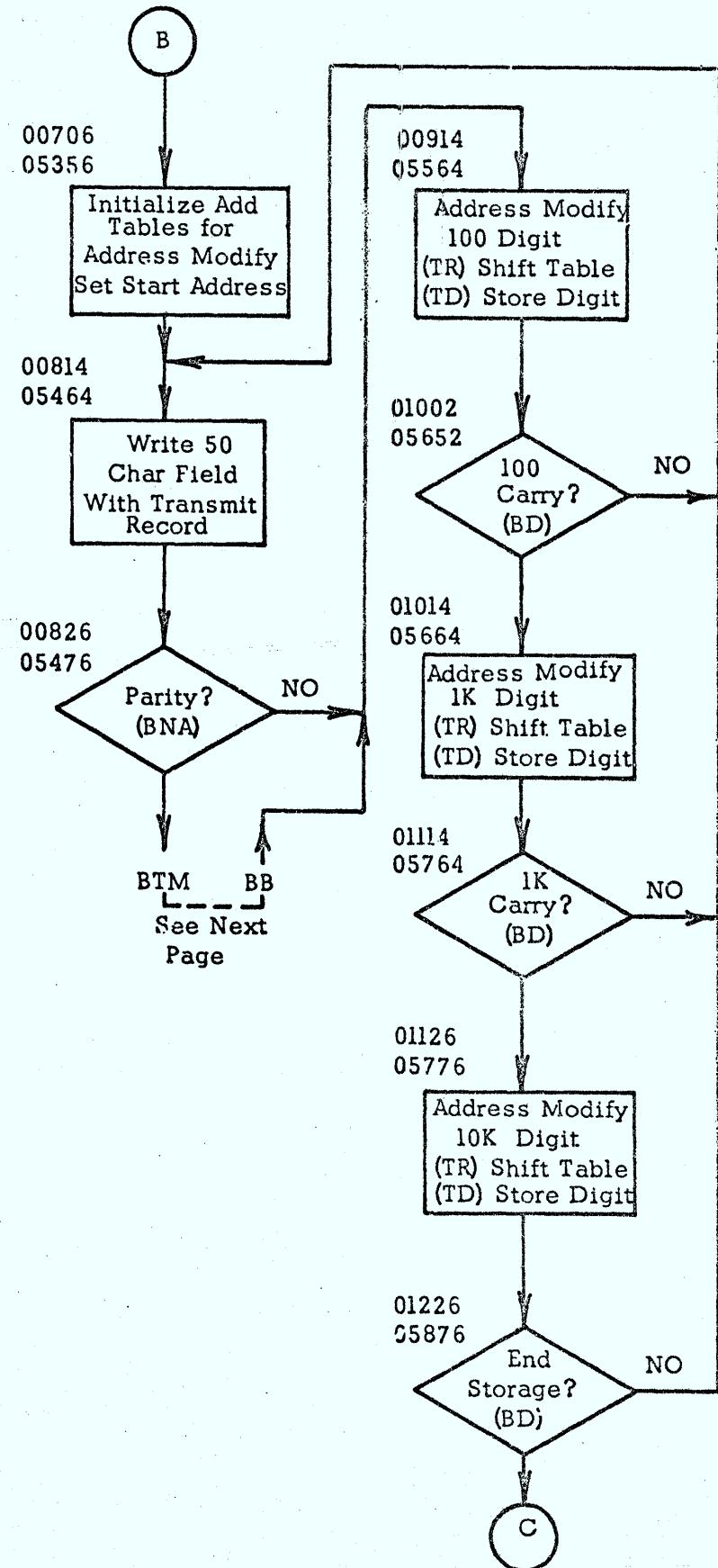
X is error char XXXXX is error ADR (Even Address)
X is error char XXXXX is error ADR (Odd Address)

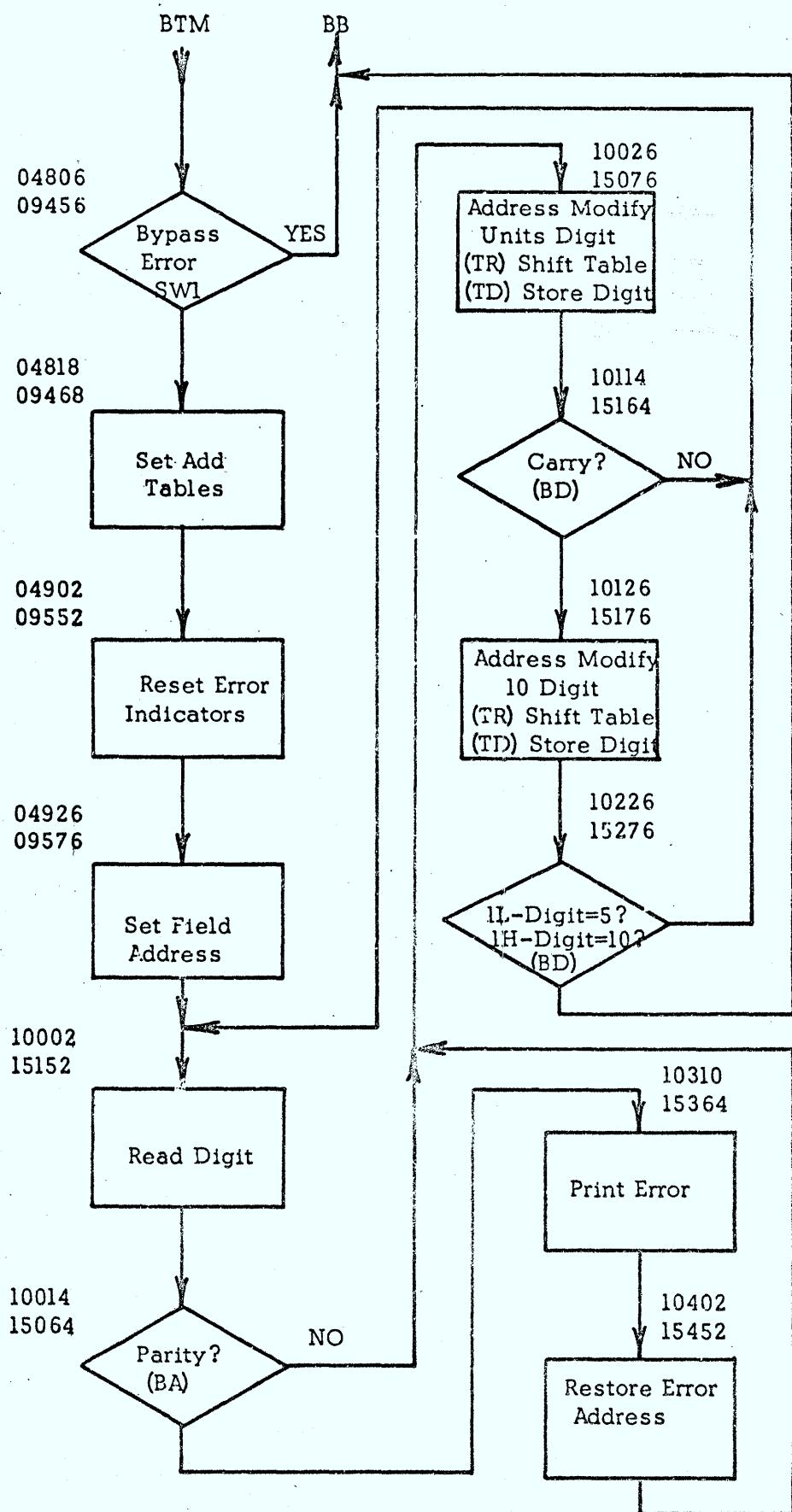
In the Compliment and Restore phase the characters from both even and odd addresses are always printed along with the even address on a single line.

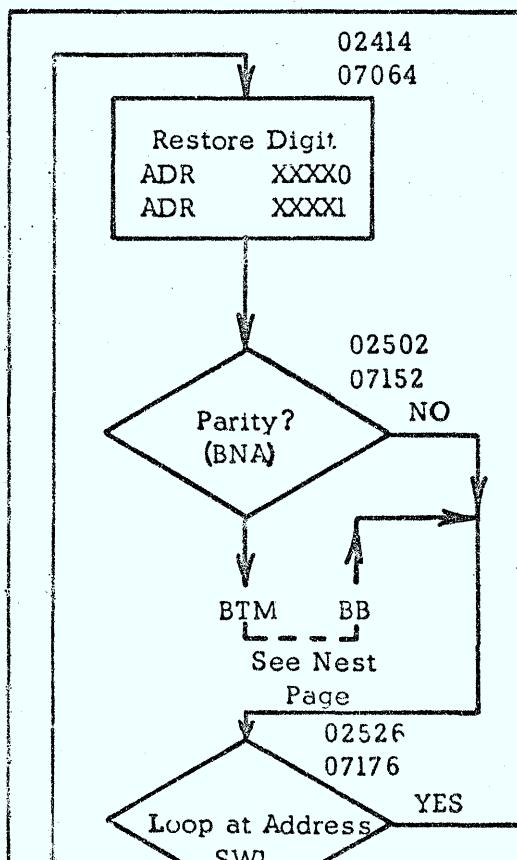
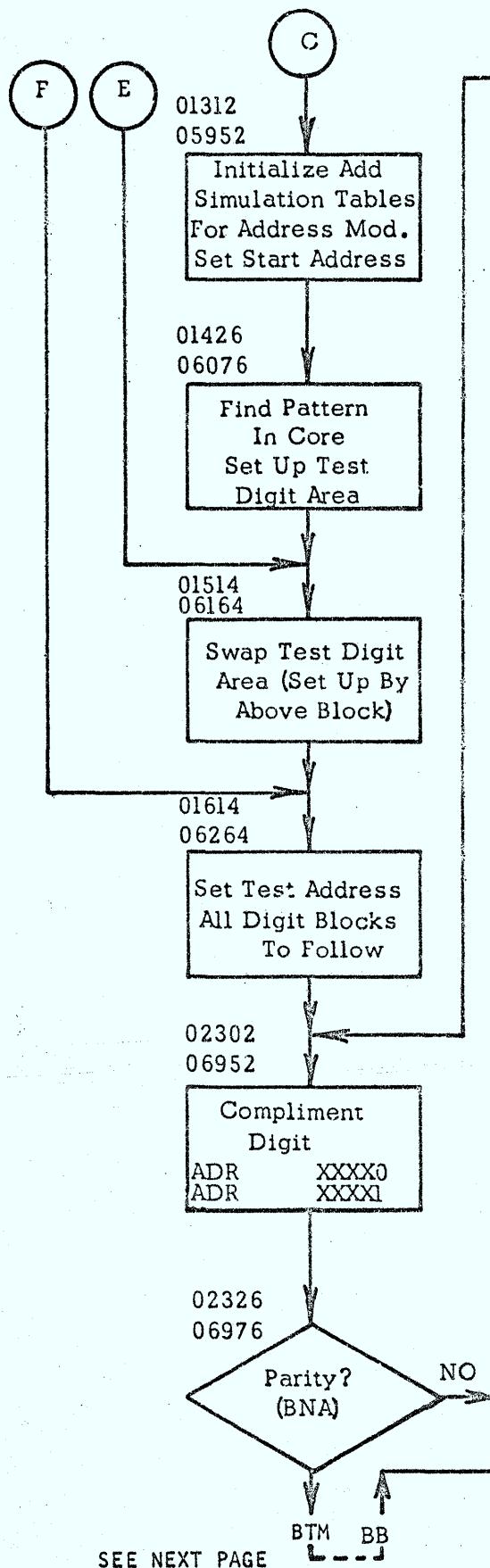
XX is error char XXXXX is error ADR (Even Address)

Both characters will be indicated as out of parity regardless of the failure.



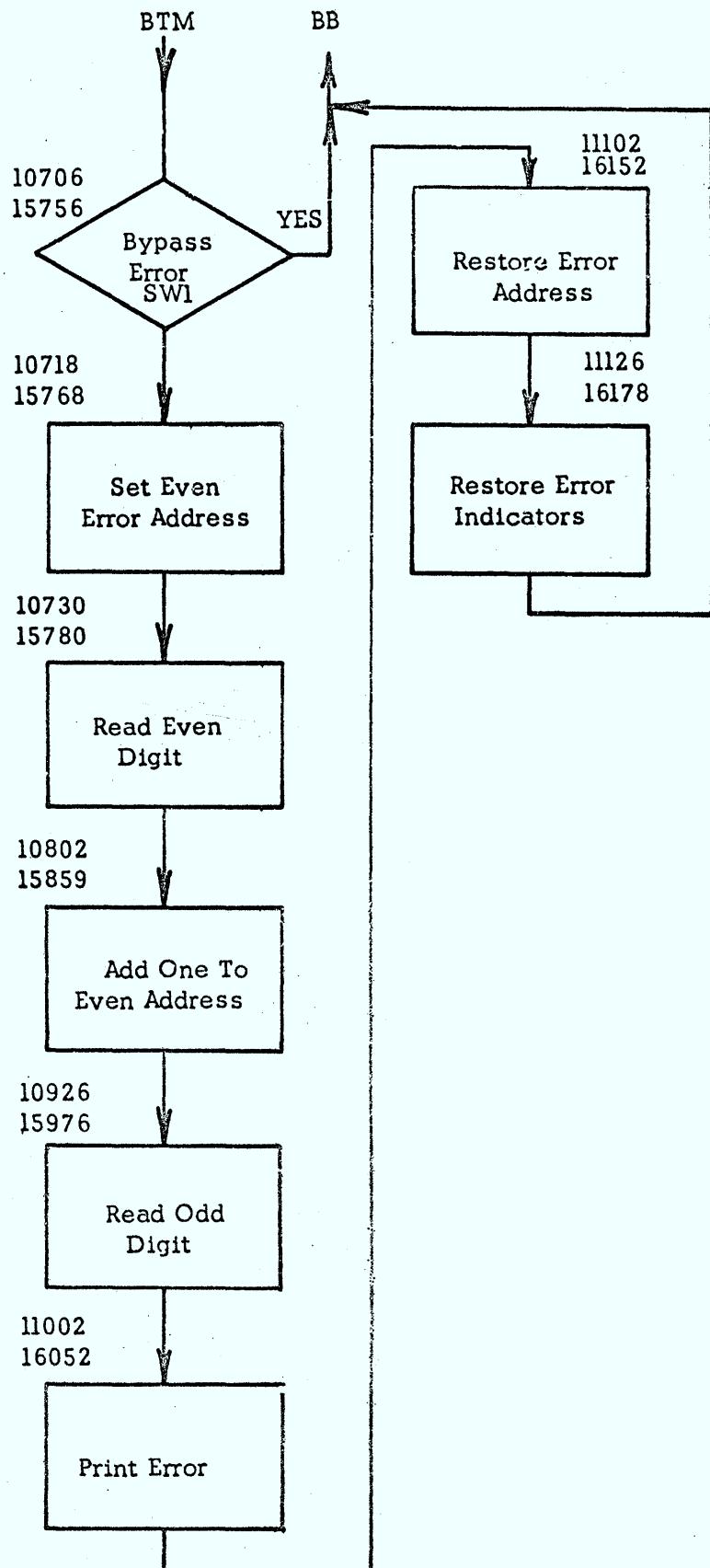


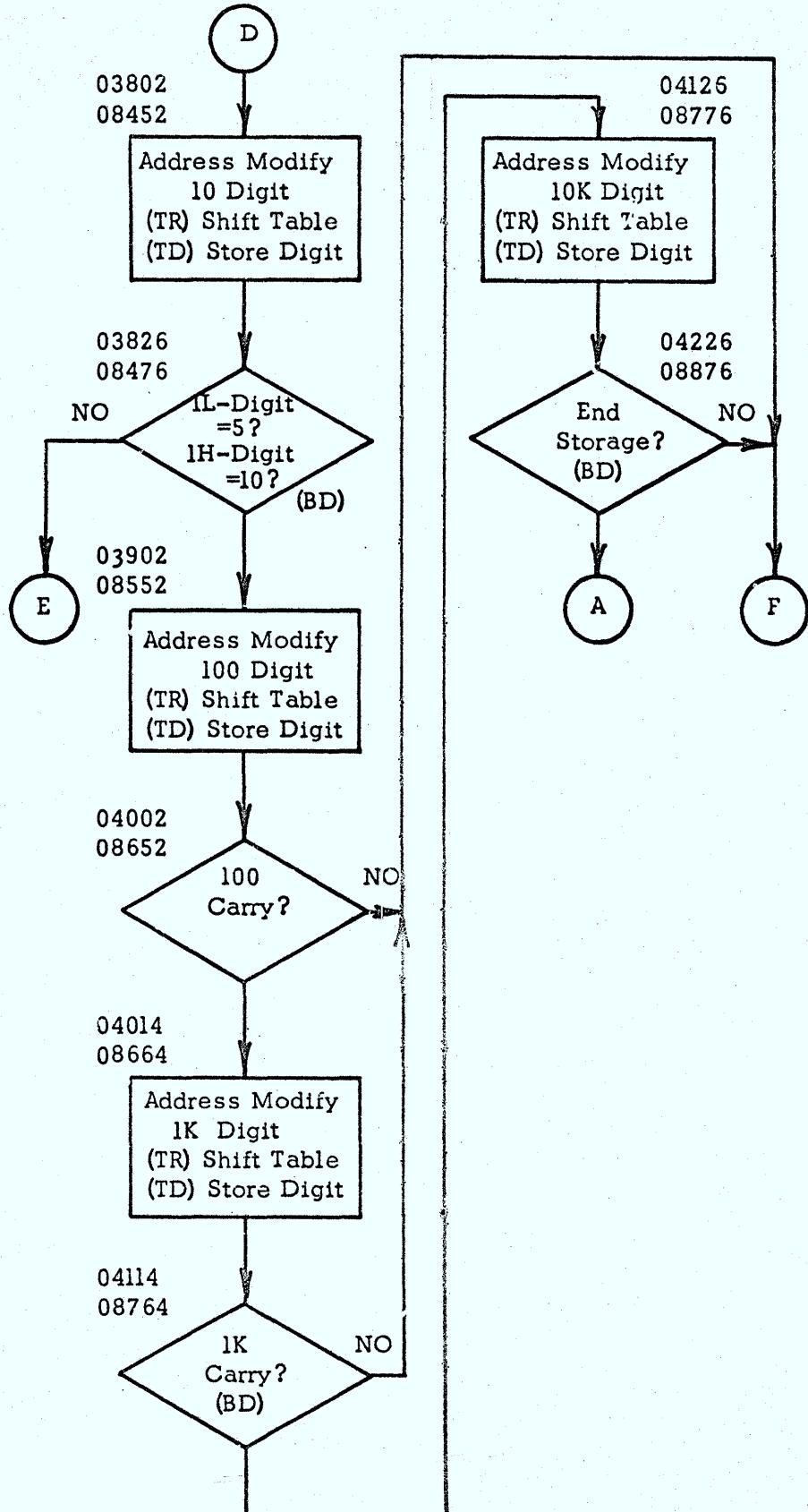




Repeat Above Routine For Addresses 0 & 1 (Last Five Blocks) Four Times. Once for Each of the Remaining Consecutive Pairs of Units Digits. (2, 3), (4, 5), (6, 7), and (8, 9). Blocks Omitted Here to Save Space.







* DTX05L
 * 1620 MEMORY TEST 20K MACHINE
 *
 *
 * THIS FIRST PART KEEPS TRACK OF WHICH PATTERN IS
 * TO BE WRITTEN NEXT AND GENERATES SAME IN LOWEST
 * TESTED AREA IN STORAGE 00-49 FOR 1L 50-99 FOR 1H
 *
 05052 DORG 5052
 05052 16 05355 -0000 TFM RTCF,,, RESET ROUTINE COUNT
 05064 49 09352 00000 B HEAD
 05076 43 09252 05354 RCB BD EXIT,RTCF-1,, EXIT
 05088 49 05152 00000 B RCC
 05152 DORG *+53
 05162 43 05252 05355 RCC BD RCD,RTCF,, BR PASS TWO 8877
 05164 31 00000 08952 TR ,TP1,, SET UP
 05176 31 00026 08954 TR 26,TP1+2,, 7788 TEST
 05188 17 05356 -0001 BTM IA,1,
 05252 DORG *+53
 05252 31 00000 09052 RCD TR ,TP2,, SET UP
 05264 31 00026 09054 TR 26,TP2+2,, 8877 TEST
 05276 17 05356 -0010 BTM IA,10,
 05288 41 00000 00000 NOP
 05351 DORG *+52
 *
 * WRITE PATTERN THROUGH STORAGE
 * IN BLOCKS OF 50 WITH TRANSMIT RECORD
 *
 05355 00005 RTCF DC 5,0,
 05356 31 09163 08981 IA TR AAD,KAD,, INITIALIZE
 05368 31 09188 08981 TR BAD,KAD,, ADD
 05380 31 09084 08995 TR CAD,KCAD,, SIMULATION TABLES
 05392 49 05452 00000 B PUA
 05452 DORG *+49
 05452 16 05470 -0000 PUA TFM WTR+6,,, SET START ADR
 05464 31 00000 00000 WTR TR ,,, WRITE FROM 00-49 INTO XXX00-XXX49
 05476 47 05564 01900 BNA AMA
 05488 49 05552 00000 B PUB
 05552 DORG *+53
 05552 27 09556 05470 PUB BT ER,WTR+6,, ERROR
 *
 * THIS ROUTINE STEPS THE TRANSMIT RECORD ADDRESS IN
 * INCREMENTS OF 100
 *
 05564 31 09162 09163 AMA TR AAD-1,AAD,, STEP 100
 05576 25 05468 09163 TD WTR+4,AAD,, SET 100
 05588 49 05652 00000 B AMB
 05652 DORG *+53
 05652 43 05464 09163 AMB BD WTR,AAD,, BR NO CARRY
 05664 31 09163 08981 TR AAD,KAD,, RESTORE 100
 05676 31 09187 09188 TR BAD-1,BAD,, STEP 1K
 05688 49 05752 00000 B AMC
 05752 DORG *+53
 05752 25 05467 09188 AMC TD WTR+3,BAD,, SET 1K
 05764 43 05464 09188 BD WTR,BAD,, BR NO CARRY
 05776 31 09188 08981 TR BAD,KAD,, RESTORE 1K
 05788 49 05852 00000 B AMD
 05852 DORG *+53
 05852 31 09083 09084 AMD TR CAD-1,CAD,, STEP 10K

05864 25 05466 09084		TD WTR+2,CAD,, SET 10K
05876 43 05464 09084		BD WTR,CAD,, NO BR ON END MEM
05888 49 05952 00000		B HKL
05952		DORG **+53
* * * COMPLEMENT AND RESTORE THE PATTERN DIGIT BY DIGIT *		
* * * FIRST SET UP STARTING ADDRESSES *		
05952 15 06957 00005	HKL	TDM WRA+5,5,, SET STARTING ADDRESS
05964 31 15789 06891		TR RAD,KRAD,, INITIALIZE
05976 31 09188 08981		TR SAD,KAD,, ADD
05988 49 06052 00000		B HKM
06052		DORG **+53
06052 31 09163 08981	HKM	TR TAD,KAD,, SIMULATION
06064 31 09084 08995		TR UAD,KCAD,, TABLES
06076 25 06963 00000		TD X,,, INITIALIZE X
06088 49 06152 00000		B HKN
06152		DORG **+53
06152 25 07263 00002	HKN	TD Y,2,, INITIALIZE Y
06164 25 06263 06963	HKP	TD XS,X,, SWAP
06176 25 06963 07263		TD X,Y,, TEST
06188 49 06252 00000		B HKQ
06252		DORG **+53
06252 15 07263 00000	HKQ	TDM Y,,, DIGITS
06263 00000	XS	DS ,*
* * * THE SUCCEEDING ARRAY OF TRANSMIT FIELD INSTRUCTIONS ARE USED TO UPDATE THE CURRENT ADDRESS IN THE FOLLOWING COMPLEMENT AND-RESTORE ROUTINE *		
06264 26 06969 06957	HKQ1	TF WRB+5,WRA+5,, SET ADR
06276 26 07069 06957		TF WRE+5,WRA+5,,
06288 49 06352 00000		B HKS
06352		DORG **+53
06352 26 07257 06957	HKS	TF WRG+5,WRA+5,,
06364 26 07081 06957		TF WRF+5,WRA+5,,
06376 26 07269 06957		TF WRH+5,WRA+5,,
06388 49 06452 00000		B HKU
06452		DORG **+53
06452 26 07557 06957	HKU	TF WRN+5,WRA+5,,
06464 26 07369 06957		TF WRL+5,WRA+5,,
06476 26 07381 06957		TF WRM+5,WRA+5,,
06488 49 06552 00000		B HKV
06552		DORG **+53
06552 26 07569 06957	HKV	TF WRP+5,WRA+5,,
06564 26 07669 06957		TF WRS+5,WRA+5,,
06576 26 07681 06957		TF WRT+5,WRA+5,,
06588 49 06652 00000		B HKW
06652		DORG **+53
06652 26 07857 06957	HKW	TF WRU+5,WRA+5
06664 26 07869 06957		TF WRV+5,WRA+5
06676 26 07969 06957		TF WRY+5,WRA+5,,
06688 49 06752 00000		B HKY
06752		DORG **+53
06752 26 08157 06957	HKY	TF WRAA+5,WRA+5,,
06764 26 08169 06957		TF WRBA+5,WRA+5,,
06776 26 07981 06957		TF WRZ+5,WRA+5,,
06788 49 06852 00000		B HKZ

06852		DORG *+53			
06852 26	08269	06957	HKZ	TF WREA+5,WRA+5,,	
06864 26	08281	06957		TF WRFA+5,WRA+5,,	
06876 49	06952	00000		B WRA	
06886	00000		TST	DS ,--1	
06888 41	00000	00000		NOP	
06891	00007		KRAD	DSC 7,012340*,*-8	
06891	00000		KEAD	DS ,KRAD	
06952		DORG *+53			
*					
* NOW WE ACTUALLY COMPLEMENT AND RESTORE THE TESTED DIGIT					
* THE BIG LOOP IS USED TO SPEED UP THE PROGRAM WHICH					
* WOULD BE IMPOSSIBLY SLOW WITH A ONE ADDRESS AT A TIME LOOP					
*					
06952 15	-0000	00000	WRA	TDM 0,,2,	WR COMP 0
06963	00000		X	DS ,*	
06964 25	00001	06963	WRB	TD 1,X,,	WR COMP 1
06976 47	07064	01900		BNA WRE	
06988 49	07052	00000		B WRC1	
07052		DORG *+53			
07052 27	15856	06958	WRC1	BT ERR,WRA+6,,	ERROR
07064 25	00000	07263	WRE	TD 0,Y,,	RESTORE 0
07076 25	00001	07263	WRF	TD 1,Y,,	RESTORE 1
07088 49	07152	00000		B WRF1	
07152		DORG *+52			
07152 47	07176	01900	WRF1	BNA WRF2	
07164 27	15856	07070		BT ERR,WRE+6,,	ERROR
07176 46	06952	00200	WRF2	BC2 WRA,,,	LOOP
07188 49	07252	00000		B WRG	
07252		DORG *+53			
07252 15	00002	07263	WRG	TDM 2,Y,,	WR COMP 2
07263	00000		Y	DS ,*	
07264 25	00003	07263	WRH	TD 3,Y,,	WR COMP 3
07276 47	07364	01900		BNA WRL	
07288 49	07352	00000		B WRH1	
07352		DORG *+53			
07352 27	15856	07258	WRH1	BT ERR,WRG+6,,	ERROR
07364 25	00002	06963	WRL	TD 2,X,,	RESTORE 2
07376 25	00003	06963	WRM	TD 3,X,,	RESTORE 3
07388 49	07452	00000		B WRM1	
07452		DORG *+53			
07452 47	07476	01900	WRM1	BNA WRM2	
07464 27	15856	07370		BT ERR,WRL+6,,	ERROR
07476 46	07252	00200	WRM2	BC2 WRG,,,	LOOP
07488 49	07552	00000		B WRN	
07552		DORG *+53			
07552 25	00004	06963	WRN	TD 4,X,,	WR COMP 4
07564 25	00005	06963	WRP	TD 5,X,,	WR COMP 5
07576 47	07664	01900		BNA WRS	
07588 49	07652	00000		B WRP1	
07652		DORG *+53			
07652 27	15856	07558	WRP1	BT ERR,WRN+6,,	ERROR
07664 25	00004	07263	WRS	TD 4,Y,,	RESTORE 4
07676 25	00005	07263	WRT	TD 5,Y,,	RESTORE 5
07688 49	07752	00000		B WRT1	
07752		DORG *+52			
07752 47	07776	01900	WRT1	BNA WRT2	
07764 27	15856	07670		BT ERR,WRS+6,,,	ERROR
07776 46	07552	00200	WRT2	BC2 WRN,,,	LOOP

07788	49	07852	00000	B	WRU	
07852				DORG	*+53	
07852	25	00006	07263	WRU	TD 6,Y,,	WR COMP 6
07864	25	00007	07263	WRV	TD 7,Y,,	WR COMP 7
07876	47	07964	01900	BNA	WRY	
07888	49	07952	00000	B	WRV1	
07952				DORG	*+53	
07952	27	15856	07858	WRV1	BT ERR,WRU+6,,	ERROR
07964	25	00006	06963	WRY	TD 6,X,,	RESTORE 6
07976	25	00007	06963	WRZ	TD 7,X,,	RESTORE 7
07988	49	08052	00000	B	WRZ1	
08052				DORG	*+53	
08052	47	08076	01900	WRZ1	BNA WRZ2	
08064	27	15856	07970		BT ERR,WRY+6,,	ERROR
08076	46	07852	00200	WRZ2	BC2 WRU,,,	LOOP
08088	49	08152	00000	B	WRAA	
08152				DORG	*+53	
08152	25	00008	06963	WRAA	TD 8,X,,	WR COMP 8
08164	25	00009	06963	WRBA	TD 9,X,,	WR COMP 9
08176	47	08264	01900	BNA	WREA	
08188	49	08252	00000	B	WRBA1	
08252				DORG	*+53	
08252	27	15856	08158	WRBA1	BT ERR,WRAA+6,,	ERROR
08264	25	00008	07263	WREA	TD 8,Y,,	RESTORE 8
08276	25	00009	07263	WRFA	TD 9,Y,,	RESTORE 9
08288	49	08352	00000	B	WRFA1	
08352				DORG	*+53	
08352	47	08376	01900	WRFA1	BNA WRFA2	
08364	27	15856	08270		BT ERR,WREA+6,,	ERROR
08376	46	08152	00200	WRFA2	BC2 WRAA,,,	LOOP
08388	49	08452	00000	B	ASRA	
08452				DORG	*+53	
* THIS SECTION STEPS THE ADDRESS OF THE COMPLEMENT AND RESTORE LOOP						
08452	31	15788	15789	ASRA	TR RAD-1,RAD,,	STEP TENS
08464	25	06957	15789		TD WRA+5,RAD,,	SET TENS
08476	43	06164	15789	ASRB	BD HKP,RAD,,	BR NO CARRY
08488	49	08552	00000		B ASRB	
08552				DORG	*+53	
08552	31	15789	06891	ASRB	TR RAD,KRAD,,	RESTORE
08564	31	09187	09188		TR SAD-1,SAD,,	STEP 100
08576	25	06956	09188	ASRC	TD WRA+4,SAD,,	SET 100
08588	49	08652	00000		B ASRC	
08652				DORG	*+53	
08652	43	06264	09188	ASRC	BD HKQ1,SAD,,	BR NO CARRY
08664	31	09188	08931		TR SAD,KAD,,	RESTORE 100
08676	31	09162	09163	ASRD	TR TAD-1,TAD,,	STEP 1K
08688	49	08752	00000		B ASRD	
08752				DORG	*+53	
08752	25	06955	09163	ASRD	TD WRA+3,TAD,,	SET 1K
08764	43	06264	09163		BD HKQ1,TAD,,	BR NO CARRY
08776	31	09163	08981	ASRE	TR TAD,KAD,,	RESTORE 1K
08788	49	08852	00000		B ASRE	
08852				DORG	*+53	
08852	31	09083	09084	ASRE	TR UAD-1,UAD,,	STEP 10K
08864	25	06954	09084		TD WRA+2,UAD,,	SET 10K
08876	43	06264	09084	ASRE	BD HKQ1,UAD,,	BR NO CARRY

08888 49 05076 00000		B RCB	
08952		DORG **53	
* * CONSTANTS AND ARITHMETIC WORKING AREAS *			
08952 00027	TP1	DSS 27,,	FIELD OF 7788
08951		DORG **27	
08954 00004		DC 4,-8778	
08958 00004		DC 4,-8778	
08962 00004		DC 4,-8778	
08966 00004		DC 4,-8778	
08970 00004		DC 4,-8778	
08974 00004		DC 4,-8778	
08978 00004		DC 4,877'	
08981 00013	KAD	DSC 13,012345678901',TP1+29	
08995 00004	KCAD	DSC 4,010',TP1+43	
08997 00002		DC 2,-10,KCAD+2	
09052		DORG TP1+100	
09052 00027	TP2	DSS 27,,	FIELD OF 8877
09052		DORG **26	
09052 00001		DC 1,8	
09056 00004		DC 4,-8778	
09060 00004		DC 4,-8778	
09064 00004		DC 4,-8778	
09068 00004		DC 4,-8778	
09072 00004		DC 4,-8778	
09076 00004		DC 4,-8778	
09078 00002		DC 2,8'	
09163 00013	AAD	DSS 13,TP2+111,	100 ADD WORK AREA
09188 00012	BAD	DSS 12,AAD+25,	1K ADD WORK AREA
09084 00004	CAD	DSS 4,TP2+32,	10K ADD WORK AREA
09163 00000	TAD	DSS ,AAD	
09188 00000	SAD	DSS ,BAD	
09084 00000	UAD	DSS ,CAD	
09252		DORG AAD+89	
* * PASS COUNTER *			
09252 39 09453 00100	EXIT	WATY PCNT,,,	PRINT PASS COMPLETE
09264 46 05052 00400		BC4 RCA	
09276 36 00000 00500		RNCD ,,,	READ NEXT PROGRAM
09288 49 00000 00000		B	
09352		DORG EXIT+100	
09352 34 00000 00102	HEAD	RCTY	
09364 39 09481 00100		WATY HD,,,	PRINT FALSE 1
09376 34 00000 00102		RCTY	
09388 49 05076 00000		B RCB	
09452		DORG HEAD+100	
09453 00014	PCNT	DAC 14,PASS COMPLETE',	
09481 00007	HD	DAC 7,DTX05L',	
09551		DORG PCNT+98	
* * ERROR ROUTINE- ERROR WHILE WRITING PATTERN *			
09555 00005	ERRX	DC 5,0,	
09556 46 15388 00100	ER	BC1 BB,,,	BYPASS ERROR
09568 31 16587 08981		TR DAD,KAD,,	SET ADD CONSTANTS
09580 31 15789 06891		TR EAD,KEAD	
09592 49 15052 00000		B ERA	

15052		DORG 15052
15052 47 15064	51655	ERA BNI ERA1,51655,, RESET MBR-E CHECK IND
15064 47 15076	51755	ERA1 BNI ERA2,51755,, RESET MBR-O CHECK IND
15076 26 15163	09555	ERA2 TF CTD+11,ERRX,, SET ERROR FIELD ADDRESS
15088 49 15152	00000	B CTD
15152		DORG *+53
15152 25 16688	-0000	CTD TD PRC,,7, READ DIGIT
15164 46 15452	01900	BA ERD,,, ERROR
15176 31 16586	16587	CTD2 TR DAD-1,DAD,, STEP UNIT ADR
15188 49 15252	00000	B ERB
15252		DORG *+53
15252 25 15163	16587	ERB TD CTD+11,DAD
15264 43 15152	16587	BD CTD,DAD
15276 31 16587	08981	TR DAD,KAD,, RESTORE UNITS
15288 49 15352	00000	B ERC
15352		DORG *+53
15352 31 15788	15789	ERC TR EAD-1,EAD,, STEP TENS
15364 25 15162	15789	TD CTD+10,EAD
15376 43 15152	15789	BD CTD,EAD,, BR NO CARRY
15388 42 00000	00000	BB BB,,, RETURN
15452		DORG *+53
15452 26 16686	15163	ERD TF PRA+4,CTD+11,,SET ERROR ADDRESS
15464 38 16688	00100	WNTY PRC,,, PRINT ERROR CHAR
15476 26 15558	15163	TF ERE+6,CTD+11
15488 49 15552	00000	B ERE
15552		DORG *+53
15552 25 0-000	16688	ERE TD ,PRC,3, RESTORE ADR
15564 47 15576	50755	EER1 BNI ERE1,50755,, RESET WRITE CHECK IND
15576 47 15652	51755	BNI ERF,51755,, RESET MBR-U CHECK IND
15588 49 15652	00000	B ERF
15652		DORG *+53
15652 47 15664	51655	ERF BNI ERF1,51655,, RESET MBR-E CHECK IND
15664 39 16753	00100	ERF1 WATY PRB
15676 38 16682	00100	WNTY PRA,,, PRINT ERROR ADR
15688 49 15752	00000	B ERG
15752		DORG *+53
15752 39 16653	00100	ERG WATY PR
15764 34 00000	00102	RCTY
15776 49 15176	00000	B CTD2
15789 00007		EAD DSS 7,*+2
15789 00000		RAD DSS ,EAD
15788 41 00000	00000	NOP
15851		DORG *+52
*	*	ERROR ROUTINE - ERROR DURING COMPLIMENT OR RESTORE
*	*	ERRY DC 5,0,
15855 00065		ERR BC1 RET,,, BYPASS ERROR
15856 46 16564	00100	TF ERR1+11,ERRY,,SET EVEN ERROR ADDRESS
15868 26 15891	15855	ERR1 TD PRJ,,, READ EVEN DIGIT
15880 25 15973	00000	B ERRJ
15892 49 15952	00000	DORG *+48
15951		ERRJ TD ERRK+11,ERRY,,SET ADD TABLE ADDRESS
15952 25 16063	15855	B ERRK
15964 49 16052	00000	PRJ DSS 2,*+2
15973 00002		DC 1,*,*
15975 00001		NOP
15976 41 00000	00000	NOP
15988 41 00000	00000	NOP
15990 00009		KHAD DSC 9,123456789,*-9

16052		DORG *+53
16052 25	15855 15990	TD ERRY,KHAD,, READ TABLE
16064 26	16087 15855	TF ERK1+11,ERRY,,SET ODD ERROR ADDRESS
16076 25	15974 00000	ERK1 TD PRJ+1,,, READ ODD DIGIT
16088 49	16152 00000	B ERRL
16152		DORG *+53
16152 38	15973 00100	WNTY PRJ,,, PRINT ERROR DIGITS
16164 26	16258 15891	TF ERM+6,ERR1+11
16176 26	16270 16087	TF ERMI+6,ERK1+11
16188 49	16252 00000	B ERM
16252		DORG *+53
16252 25	00000 15973	TD vPRJ,, RESTORE ERROR DIGITS
16264 25	00000 15974	TD vPRJ+1
16276 47	16352 50755	BNI ERRN,50755,, RESTORE CHECK IND
16286 49	16352 00000	B ERRN,
16352		DORG *+53
16352 47	16364 51755	BNI ERN1,51755
16364 47	16376 51655	BNI ERN2,51655
16376 59	16753 00100	WATY PRB
16388 49	16452 00000	B ERRP
16452		DORG *+53
16452 26	16573 15891	TF ERY1+4,ERR1+11,,SET ERROR ADDRESS FOR PRINT
16464 38	16569 00100	WNTY ERY1,,, PRINT ERROR ADR
16476 39	16653 00100	WATY PR
16488 49	16552 00000	B ERRQ
16552		DORG *+53
16552 34	00000 00102	RCTY
16564 42	00000 00000	REY BB *** RETURN
		*
		*
		*
		CONSTANTS AND ARITHMETIC WORKING AREAS
16569	00006	ERY1 DSC 6,*,-6
16576 41	00000 00000	NOP
16587	00012	DAD DSS 12,*
16588 41	00000 00000	NOP
16652		DORG ERRQ+100
16653	00015	PR DAC 15, IS ERROR ADR *,
16682	00005	PRA DSS 5
16687	00001	DC 1,*
16688	00001	PRC DSS 1
16689	00001	DC 1,*
16752		DORG PR+99
16753	00014	PRB DAC 14, IS ERR CHAR *,
05052		DEND 5052

DT X05L 80/80 LIST

360007200500360020100500440001200276260005900274250001100000260009000269	-0000
26000950026431000000200260011400274250000000011490001200000	-0001
250726300002250626306963250696307263490625200000*	0-1-6152-6200 -0002
1605355-00004909352000043092520535490515200000*	0-1-5052-5100 -0003
430525205355310000089523100026089541705356-0001*	0-1-5152-5200 -0004
3100000090523100026090541705356-0010410000000000*	0-1-5252-5300 -0005
-0000#	1-1-5351-5356 -0006
310916308981310918808981310908408995490545200000*	0-1-5356-5404 -0007
1605470-000031000000000470556401900490555200000*	0-1-5452-5500 -0008
270955605470310916209163250546809163490565200000*	0-1-5552-5600 -0009
430546409163310916308981310918709188490575200000*	0-1-5652-5700 -0010
250546709188430546409188310918808981490585200000*	0-1-5752-5800 -0011
310908309084250546609084430546409084490595200000*	0-1-5852-5900 -0012
150695700005311578906891310918808981490605200000*	0-1-5952-6000 -0013
310916308981310908408995250696300000490615200000*	0-1-6052-6100 -0014
250726300002250626306963250696307263490625200000*	0-1-6152-6200 -0015
15072630000260696906957260706906957490635200000*	0-1-6252-6300 -0016
260725706957260708106957260726906957490645200000*	0-1-6352-6400 -0017
260755706957260736906957260738106957490655200000*	0-1-6452-6500 -0018
260756906957260766906957260768106957490665200000*	0-1-6552-6600 -0019
260785706957260786906957260796906957490675200000*	0-1-6652-6700 -0020
260815706957260816906957260798106957490685200000*	0-1-6752-6800 -0021
260826906957260828106957490695200000410000000000*	0-1-6852-6900 -0022
012340#	1-1-6891-6898 -0023
15-000000000250000106963470706401900490705200000*	0-1-6952-7000 -0024
27158560695825000007263250000107263490715200000*	0-1-7052-7100 -0025
470717601900271585607070460695200200490725200000*	0-1-7152-7200 -0026
150000207263250000307263470736401900490735200000*	0-1-7252-7300 -0027
271585607258250000206963250000306963490745200000*	0-1-7352-7400 -0028
470747601900271585607370460725200200490755200000*	0-1-7452-7500 -0029
250000406963250000506963470766401900490765200000*	0-1-7552-7600 -0030
271585607558250000407263250000507263490775200000*	0-1-7652-7700 -0031
470777601900271585607670460755200200490785200000*	0-1-7752-7800 -0032
250000607263250000707263470796401900490795200000*	0-1-7852-7900 -0033
271585607858250000606963250000706963490805200000*	0-1-7952-8000 -0034
470807601900271585607970460785200200490815200000*	0-1-8052-8100 -0035
250000806963250000906963470826401900490825200000*	0-1-8152-8200 -0036
271585608158250000807263250000907263490835200000*	0-1-8252-8300 -0037
470837601900271585608270460815200200490845200000*	0-1-8352-8400 -0038
311578815789250695715789430616415789490855200000*	0-1-8452-8500 -0039
311578906891310918709188250695609188490865200000*	0-1-8552-8600 -0040
430626409188310918808981310916209163490875200000*	0-1-8652-8700 -0041
250695509163430626409163310916308981490885200000*	0-1-8752-8800 -0042
310908309084250695409084430626409084490507600000*	0-1-8852-8900 -0043
Q77QQ77QQ77QQ77QQ77QQ77QQ77#	1-1-8951-8979 -0044
012345678901#	1-1-8981-8994 -0045
010#	1-1-8995-8999 -0046
J-#	1-1-8996-8998 -0047
QQ77QQ77QQ77QQ77QQ77QQ77QQ77#	1-1-9052-9079 -0048
3909453001004605020040036000000050049000000000000*	0-1-9252-9300 -0049
34000000010239094810010034000000102490507600000*	0-1-9352-9400 -0050
N74162620043565457534563450#	1-1-9452-9480 -0051
M463677075530#	1-1-9480-9494 -0052
-0000#	1-1-9551-9556 -0053
46153800100311658708981311578906891491505200000*	0-1-9556-9604 -0054
471506451655471507651755261516309555491515200000*	0-1J5052J5100 -0055
2516688-0000461545201900311658616587491525200000*	0-1J5152J5200 -0056
251516316587431515216587311658708981491535200000*	0-1J5252J5300 -0057

31157881578925151621578943151521578942000000000000‡	0-1J5352J5400 -0058
261668615163381668800100261555815163491555200000‡	0-1J5452J5500 -0059
250-00016688471557650755471565251755491565200000‡	0-1J5552J5600 -0060
471566451655391675300100381668200100491575200000‡	0-1J5652J5700 -0061
391665300100340000000102491517600004100000000000‡	0-1J5752J5800 -0062
-0000‡	1-1J5851J5856 -0063
46165640010026158911585525159730000491595200000‡	0-1J5856J5904 -0064
251606315855491605200000‡	0-1J5952J5976 -0065
‡	1-1J5975J5976 -0066
41000000000410000000000‡	0-1J5976J6000 -0067
123456789‡	1-1J5990J5999 -0068
25158551599026160871585525159740000491615200000‡	0-1J6052J6100 -0069
381597300100261625815891261627016087491625200000‡	0-1J6152J6200 -0070
250000015973250000015974471635250755491635200000‡	0-1J6252J6300 -0071
471636451755471637651655391675300100491645200000‡	0-1J6352J6400 -0072
261657315891381656900100391665300100491655200000‡	0-1J6452J6500 -0073
34000000010242000000000‡	0-1J6552J6576 -0074
00000‡	1-1J6569J6575 -0075
41000000000410000000000‡	0-1J6576J6600 -0076
-0496200455959565900414459000‡	1-1J6652J6682 -0077
‡	1-1J6687J6688 -0078
‡	1-1J6689J6690 -0079
-04962004559590043484159000‡	1-1J6752J6780 -0080
00000 L60000005004900000‡	-8-0096-0115 -0081
360010000500360017200500360024400500360031600500360000000500	-0082
00000000000010203040002040680003060902100408021610050015102006021814200‡	-0083
704112820080614223009081726300000000005060708090012141618151811242720242‡	-0084
822363520353045403632484455324946536048465462754453627180123456789123456‡	-0085
789-23456789-J3456789-JK456789-JKL56789-JKLM6789-JKLMN789-JKLMN089-JKLMN‡	-0086
M800000000049-50520P9-JKLMN0PQ‡ L10038800019M9000000000M900036C0000	-0087

DT X05H 80/80 LIST

CU04– Additional Core Diagnostic


```
//=====
//  
// CU04 - Additional Core Diagnostic  
//  
// Program Switch settings:  
//  
// PS1:   ON - Bypass error type out  
//         OFF - Type out routine number on error  
// PS2:   ON - Loop in routine  
//         OFF - Continue to next routine  
// PS3:   ON - Stop on error  
//         OFF - Do not stop on error, continue  
// PS4:   not used  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY    - STOP  
// I/O       - STOP  
// O'FLOW    - PROGRAM  
//  
// Start addresses:  
//  
// 00828 - Full test  
//  
// Directions:  
//  
// 1. Load CU04 diagnostic  
// 2. Press START  
// 3. Press START  
// 4. Type <flag 6> 0  
// 5. Press RELEASE-START  
//=====
```


Sample Output - CU04

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR ADDITIONAL MEMORY TEST. THEN START.

START ROUTINES. ETOS FOLLOW. KEY IN MEM CONSTS RELEASE. START. ~~60R\$~~

39976Ø123456789‡1Ø399989 3999894ØØØ4

ABCDEFGHIJKLMNPQRSTUVWXYZ.)+\$:-/, (=Ø123456789

59976Ø123456789‡1Ø599989 59998949ØØ828

Ø123456789.)+\$:-/, (=ØABCDEFGHIJKLMNPQRSTUVWXYZ

TEST ROUTINES COMPLETED. IF SW 1 OFF AND NO ROUTINE NOS TYPED OUT, MACHINE PERFORMED TESTS PROPERLY.

// CU04 - Additional Core diagnostic

00000: 04 09 00 00 08 02 08 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 // "I # " 00100: 00 00 00 00 00 00 00 00 00 00 00 01 00 02 00 03 00 04 00 // " +- " 00120: 00 00 02 00 04 00 06 00 08 00 00 00 03 00 06 00 09 00 02 01 // " - /" 00140: 00 00 04 00 08 00 02 01 06 01 00 00 05 00 00 01 05 01 00 02 // " / - J " 00160: 00 00 06 00 02 01 08 01 04 02 00 00 07 00 04 01 01 02 08 02 // " / B 0A " 00180: 00 00 08 00 06 01 04 02 02 03 00 00 09 00 08 01 07 02 06 03 // " B, 2T" 00200: 00 00 00 00 00 00 00 00 00 00 05 00 06 00 07 00 08 00 09 00 // " - 0 " 00220: 00 01 02 01 04 01 06 01 08 01 05 01 08 01 01 02 04 02 07 02 // " /A J B2" 00240: 00 02 04 02 08 02 02 03 06 03 05 02 00 03 05 03 00 04 05 04 // " B ,T K.L)M" 00260: 00 03 06 03 02 04 08 04 04 05 05 03 02 04 09 04 06 05 03 06 // ".T(E L(V " 00280: 00 04 08 04 06 05 04 06 02 07 05 04 04 05 03 06 02 07 01 08 // ") VF ME " 00300: 00 01 02 03 04 05 06 07 08 09 01 02 03 04 05 06 07 08 09 10 // ",EX @08 " 00320: 02 03 04 05 06 07 08 09 10 11 03 04 05 06 07 08 09 10 11 12 // ",EX @08 " 00340: 04 05 06 07 08 09 10 11 12 13 05 06 07 08 09 10 11 12 13 14 // "EX , O8 @" 00360: 06 07 08 09 10 11 12 13 14 15 07 08 09 10 11 12 13 14 15 16 // "X ,E 8 @O" 00380: 08 09 10 11 12 13 14 15 16 17 09 10 11 12 13 14 15 16 17 18 // " ,EX @08" 00400: 0A 00 00 00 00 00 00 05 02 04 05 06 08 00 00 04 09 05 05 // " K EY IN" 00420: 00 00 05 04 04 05 05 04 00 00 04 03 05 06 05 05 06 02 06 03 // " MEM CONST" 00440: 06 02 00 00 05 09 04 05 05 03 04 05 04 01 06 02 04 05 00 03 // "S REL EASE." 00460: 00 00 06 02 06 03 04 01 05 09 06 03 00 03 00 00 00 0A 00 00 // " STAR T. # " 00480: 00 00 00 00 00 00 00 00 00 00 00 00 00 06 02 06 03 04 01 05 09 // " STAR" 00500: 06 03 00 00 05 09 05 06 06 04 06 03 04 09 05 05 04 05 06 02 // "T ROU TINES" 00520: 00 03 00 00 04 05 06 03 05 06 06 02 00 00 04 06 05 06 05 03 // ". ETO S FOL" 00540: 05 03 05 06 06 06 00 03 00 00 00 0A 00 00 00 00 00 00 00 00 // "LOW. # " 00560: 00 00 00 00 06 02 06 06 00 00 07 01 00 00 05 06 05 05 00 00 // " SW 1 ON" 00580: 00 0A 06 02 06 06 00 00 07 01 00 00 05 06 04 06 04 06 00 00 // "#SW 1 OFF" 00600: 00 0A 06 02 06 06 00 00 07 02 00 00 05 06 05 05 00 00 00 0A // "#SW 2 ON #" 00620: 06 02 06 06 00 00 07 02 00 00 05 06 04 06 04 06 00 00 00 0A // "SW 2 OFF #" 00640: 06 02 06 06 00 00 07 03 00 00 05 06 05 05 00 00 00 00 0A 06 02 // "SW 3 ON #S" 00660: 06 06 00 00 07 03 00 00 05 06 04 06 04 06 00 00 00 0A 06 02 // "W 3 O FF #S" 00680: 06 06 00 00 07 04 00 00 05 06 05 05 00 00 00 0A 06 02 06 06 // "W 4 O N #SW" 00700: 00 00 07 04 00 00 05 06 04 06 04 06 00 00 00 0A 06 02 04 05 // " 4 OF F #SE" 00720: 06 03 00 00 06 02 06 06 06 02 00 00 04 06 05 06 05 09 00 00 // "T SWS FOR" 00740: 04 01 04 04 04 04 09 06 03 04 09 05 06 05 05 04 01 05 03 // "ADDIT IONAL" 00760: 00 00 05 04 04 05 05 04 05 06 05 09 06 08 00 00 00 06 03 04 05 // " MEMO RY TE" 00780: 06 02 06 03 00 03 00 00 06 03 04 08 04 05 05 05 00 00 06 02 // "ST. T HEN S" 00800: 06 03 04 01 05 09 06 03 00 03 00 00 00 0A 00 00 00 00 00 00 00 // "TART. # " 00820: 00 00 00 00 00 00 00 00 04 06 00 00 08 05 02 00 00 01 00 00 // " F - " 00840: 04 07 00 00 08 07 06 00 00 01 00 00 03 09 00 00 05 06 05 00 // "G O- " 00860: 00 01 00 00 04 09 00 00 08 08 08 00 00 00 00 00 03 09 00 00 // " I " 00880: 05 08 03 00 00 01 00 00 04 06 00 00 09 01 02 00 00 02 00 00 // "Q F - " 00900: 04 07 00 00 09 03 06 00 00 02 00 00 03 09 00 00 06 00 03 00 // "G " 00920: 00 01 00 00 04 09 00 00 09 04 08 00 00 00 00 00 03 09 00 00 // " I " 00940: 06 02 01 00 00 01 00 00 04 06 00 00 09 07 02 00 00 03 00 00 // "S+ F - . " 00960: 04 07 00 00 09 09 06 00 00 03 00 00 03 09 00 00 06 04 01 00 // "G . U+" 00980: 00 01 00 00 04 09 00 01 00 00 08 00 00 00 00 00 03 09 00 00 // " I " 01000: 06 05 09 00 00 01 00 00 04 06 00 01 00 03 02 00 00 04 00 00 // "V F .-)" 01020: 04 07 00 01 00 05 06 00 00 04 00 00 03 09 00 00 06 07 09 00 // "G) X " 01040: 00 01 00 00 04 09 00 01 00 06 08 00 00 00 00 00 03 09 00 00 // " I " 01060: 06 09 07 00 00 01 00 00 03 09 00 00 07 01 07 00 00 01 00 00 // "Z0 10 " 01080: 04 08 00 00 00 00 00 00 00 00 00 00 03 04 00 00 00 00 00 00 // "H @ " 01100: 00 01 00 02 03 09 00 00 04 09 03 00 00 01 00 00 04 09 00 01 // " I I " 01120: 01 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 // " * " 01140: 03 09 00 00 04 00 09 00 00 01 00 00 03 06 00 01 01 03 03 00 // " \$ " 01160: 00 01 00 00 04 09 00 01 03 00 08 00 00 00 00 00 00 00 00 00 // " I " 01180: 00 00 00 00 00 00 00 10 00 00 03 00 04 01 00 01 03 01 04 // " .)+\$* " 01200: 02 00 02 01 02 03 02 04 03 03 03 04 04 01 04 02 04 03 04 04 // " -/, (= @ABCD" 01220: 04 05 04 06 04 07 04 08 04 09 05 01 05 02 05 03 05 04 05 05 // "EFGHI JKLMN"

01240: 05 06 05 07 05 08 05 09 06 01 06 02 06 03 06 04 06 05 06 06 // "OPQR STUVW"
01260: 06 07 06 08 06 09 07 00 07 01 07 02 07 03 07 04 07 05 07 06 // "XYZ01 23456"
01280: 07 07 07 08 07 09 06 09 00 0A 00 00 00 00 00 00 00 00 00 00 // "789Z#" "
01300: 00 00 00 00 00 00 00 03 01 02 09 09 05 00 00 01 01 08 08 // " "
01320: 02 04 00 01 02 08 07 03 00 00 04 09 04 07 00 01 03 08 00 00 // "(3 IG "
01340: 01 02 00 00 04 09 00 01 04 07 06 00 00 00 00 00 00 00 00 00 // " I G "
01380: 04 06 00 01 04 05 02 00 00 01 00 00 03 09 00 01 04 06 05 00 // "F E- F-"
01400: 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 00 01 // " @ "
01420: 01 08 09 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 // " @ "
01440: 03 09 02 09 09 05 01 00 00 01 00 00 04 07 00 01 04 07 06 00 // " + G G "
01460: 00 03 00 00 04 08 07 00 07 00 07 03 00 00 00 0A 04 06 00 01 // ". H00 3 #F "
01480: 03 00 08 00 00 02 00 00 04 09 00 01 05 04 08 00 00 00 00 00 // " I M "
01540: 00 00 00 00 00 00 00 00 03 01 00 09 09 05 00 02 09 09 05 00 // " "
01560: 02 04 03 00 00 04 09 01 00 00 04 09 04 07 00 01 06 03 02 00 // "() IG T-"
01580: 01 02 00 00 04 09 00 01 07 02 08 00 00 00 00 00 00 00 00 00 // " I 2 "
01620: 00 00 00 00 00 00 00 00 00 00 00 00 04 06 00 01 07 00 04 00 // " F 0 "
01640: 00 01 00 00 03 09 00 01 07 01 07 00 00 01 00 00 03 04 00 00 // " 1 0 @ "
01660: 00 00 00 00 00 01 00 02 03 09 02 09 09 05 01 00 00 01 00 00 // " + "
01680: 03 04 00 00 00 00 00 00 01 00 02 03 09 00 09 09 05 01 00 // "@ +"
01700: 00 01 00 00 04 07 00 01 07 02 08 00 00 03 00 00 04 08 07 00 // " G 2 . H0"
01720: 07 00 07 04 00 00 00 0A 04 06 00 01 05 04 08 00 00 02 00 00 // "04 #F M "
01740: 04 09 00 01 08 07 02 00 00 00 00 00 00 00 00 00 00 00 00 00 // "I - "
01760: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 03 05 06 // " CO"
01780: 05 05 06 02 06 03 06 02 00 00 04 09 05 05 04 03 05 06 05 09 // "NSTS INCOR"
01800: 05 09 00 03 00 00 05 02 04 05 06 08 00 00 04 09 05 05 00 00 // "R. KE Y IN "
01820: 04 03 05 06 05 09 05 09 00 00 04 03 05 06 05 05 06 02 06 03 // "CORR CONST"
01840: 06 06 00 00 04 01 05 05 04 04 00 00 06 02 06 03 04 01 05 09 // "W AND STAR"
01860: 04 05 06 03 00 03 00 00 00 0A 00 00 01 04 00 01 01 03 04 00 // "ET. # * \$ "
01880: 00 00 14 00 04 06 00 03 00 09 06 00 01 02 00 00 01 04 00 01 // " F. * "
01900: 01 03 04 00 00 00 16 00 04 06 00 02 00 08 08 00 01 02 00 00 // \$" F "
01920: 03 09 00 01 07 07 07 00 00 01 00 00 04 09 00 01 01 04 00 00 // " 70 I * "
01960: 00 00 00 00 00 00 00 00 16 09 07 09 07 08 07 07 07 06 07 05 // " Z 98765"
01980: 07 04 07 03 07 02 07 01 07 00 06 09 06 08 06 07 06 06 06 05 // "43210 ZYXWV"
02000: 06 04 06 03 06 02 06 01 05 09 05 08 05 07 05 06 05 05 05 04 // "UTS R QPONM"
02020: 05 03 05 02 05 01 04 09 04 08 04 07 04 06 04 05 04 04 04 03 // "LKJIH GFEDC"
02040: 04 02 04 01 03 04 03 03 02 04 02 03 02 01 02 00 01 04 01 03 // "BA@=(,/-*\$"
02060: 01 00 00 04 00 00 00 03 00 0A 00 00 00 00 00 00 00 00 00 00 // "+) .# "
02080: 00 00 00 00 00 00 00 00 03 01 04 09 09 05 00 00 01 09 06 08 // " I Y"
02100: 02 04 00 02 00 06 07 05 00 00 04 09 04 07 00 02 01 06 00 00 // "(5 IG "
02120: 01 02 00 00 04 09 00 02 02 05 06 00 00 00 00 00 00 00 00 00 // " I "
02160: 04 06 00 02 02 03 02 00 00 01 00 00 03 09 00 02 02 04 05 00 // "F , - (-"
02180: 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 00 01 // " @ "
02200: 09 06 09 00 00 01 00 00 03 04 00 00 00 00 00 00 01 00 02 // " @ "
02220: 03 09 04 09 09 05 01 00 00 01 00 00 04 07 00 02 02 05 06 00 // " I + G "
02240: 00 03 00 00 04 08 07 00 07 00 07 06 00 00 00 0A 04 06 00 02 // ". H00 6 #F "
02260: 00 08 08 00 00 02 00 00 04 09 00 02 03 00 04 00 00 00 00 00 // " I "
02300: 00 00 00 00 03 01 00 09 09 05 00 04 09 09 05 00 02 04 05 00 // ") -(-"
02320: 00 04 09 01 00 00 04 09 04 07 00 02 03 07 06 00 01 02 00 00 // ") IG "
02340: 04 09 00 02 04 07 02 00 00 00 00 00 00 00 00 00 00 00 00 00 // "I G- "
02360: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 06 00 02 // " F "
02380: 04 04 08 00 00 01 00 00 03 09 00 02 04 06 01 00 00 01 00 00 // "D F+ "
02400: 03 04 00 00 00 00 00 00 00 01 00 02 03 09 04 09 09 05 01 00 // "@ I +"
02420: 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 00 09 // " @ "
02440: 09 05 01 00 00 01 00 00 04 07 00 02 04 07 02 00 00 03 00 00 // " + G G-. "
02460: 04 08 07 00 07 00 07 03 00 00 00 0A 04 06 00 02 03 00 04 00 // "H003 #F "
02480: 00 02 00 00 04 09 00 02 06 04 00 00 00 00 00 00 00 00 00 00 // " I U "
02520: 14 01 04 02 04 03 04 04 04 05 04 06 04 07 04 08 04 09 05 01 // "ABCDE FGHIJ"
02540: 05 02 05 03 05 04 05 05 05 06 05 07 05 08 05 09 06 01 06 02 // "KLMNO PQR S"
02560: 06 03 06 04 06 05 06 06 06 07 06 08 06 09 00 00 00 03 00 04 // "TUVWX YZ .)"
02580: 01 00 01 03 01 04 02 00 02 01 02 03 02 04 03 03 03 04 07 00 // "+\$*-/, (=@0"

02600: 07 01 07 02 07 03 07 04 07 05 07 06 07 07 07 08 07 09 06 09 // "12345 6789Z"
02620: 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 // "# "
02640: 03 01 02 09 08 05 00 00 02 05 02 00 03 01 04 09 09 05 00 02 // " - I "
02660: 09 08 05 00 02 04 05 00 00 04 09 02 09 09 04 09 04 07 00 02 // " -(-) IG "
02680: 07 02 04 00 01 02 00 00 04 09 00 02 08 02 00 00 00 00 00 00 // "2 I "
02720: 00 00 00 00 04 06 00 02 07 09 06 00 00 01 00 00 03 09 00 02 // " F 9 "
02740: 08 00 09 00 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 // " @ "
02760: 03 09 02 09 08 05 01 00 00 01 00 00 03 04 00 00 00 00 00 00 // " + @ "
02780: 00 01 00 02 03 09 04 09 09 05 01 00 00 01 00 00 04 07 00 02 // " I + G "
02800: 08 02 00 00 00 03 00 00 04 08 07 00 07 00 07 08 00 00 00 0A // " . H 008 #"
02820: 04 06 00 02 06 04 00 00 00 02 00 00 04 09 00 02 08 06 08 00 // "F U I "
02860: 00 00 00 00 00 00 00 03 01 02 09 09 05 00 04 09 09 05 00 // ") -"
02880: 02 04 03 00 00 04 09 05 00 00 04 09 04 07 00 02 09 04 00 00 // " () IG "
02900: 01 02 00 00 04 09 00 03 00 03 06 00 00 00 00 00 00 00 00 // " I.. "
02940: 04 06 00 03 00 01 02 00 00 01 00 00 03 09 00 03 00 02 03 00 // "F. - . "
02960: 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 04 09 // " @ I "
02980: 09 05 01 00 00 01 00 00 03 04 00 00 00 00 00 00 01 00 02 // " + @ "
03000: 03 09 02 09 09 05 01 00 00 01 00 00 04 07 00 03 00 03 06 00 // " + G.. "
03020: 00 03 00 00 04 08 07 00 07 00 07 09 00 00 00 0A 04 06 00 02 // " . H00 9 #F "
03040: 08 06 08 00 00 02 00 00 04 09 00 03 00 09 06 00 00 00 00 00 // " I . "
03080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 10 00 04 06 00 03 // " F."
03100: 01 00 08 00 01 04 00 00 01 01 00 03 00 09 05 00 00 00 10 01 // " + * . - "
03120: 04 07 00 01 03 00 08 00 01 04 00 00 04 09 00 03 01 06 08 00 // "G * I. "
03160: 00 00 00 00 00 00 00 00 04 01 03 09 09 05 02 00 00 03 00 00 // " A - . "
03180: 03 04 00 00 00 00 00 00 00 01 00 02 03 05 03 09 09 07 06 00 // "@ "
03200: 00 01 00 00 03 04 00 00 00 00 00 00 00 01 00 01 04 09 03 09 // " @ I "
03220: 09 05 02 00 00 00 00 04 09 00 03 02 04 00 00 00 00 00 00 // " - I .("
03240: 04 01 02 05 08 05 01 00 00 03 00 00 03 04 00 00 00 00 00 // "A +. @ "
03260: 00 01 00 02 03 09 02 05 08 05 01 00 00 01 00 00 04 09 00 03 // " + I."
03280: 02 08 08 00 00 00 00 01 04 00 01 01 03 04 00 00 00 14 00 // " * \$ "
03300: 04 06 00 03 06 08 04 00 01 02 00 00 01 04 00 00 01 01 03 04 00 // "F.Y * \$ "
03320: 00 00 16 00 04 06 00 03 03 06 00 00 01 02 00 00 03 09 00 01 // " F. "
03340: 07 07 07 00 00 01 00 00 04 09 00 01 01 04 00 00 00 00 00 00 // "70 I * "
03360: 04 01 05 09 09 05 02 00 00 03 00 00 03 04 00 00 00 00 00 00 // "AR - . @ "
03380: 00 01 00 02 03 05 05 09 09 07 06 00 00 01 00 00 03 04 00 00 // " R @ "
03400: 00 00 00 00 00 01 00 01 04 09 05 09 09 05 02 00 00 00 00 00 // " I R - "
03420: 04 09 00 03 04 03 02 00 00 00 00 00 04 01 04 05 08 05 01 00 // "I.C- AE +"
03440: 00 03 00 00 03 04 00 00 00 00 00 00 00 01 00 02 03 09 04 05 // ". @ E"
03460: 08 05 01 00 00 01 00 00 04 09 00 03 06 08 04 00 00 00 00 00 // " + I .Y "
03480: 06 03 04 05 06 02 06 03 00 00 05 09 05 06 06 04 06 03 04 09 // "TEST ROUTI"
03500: 05 05 04 05 06 02 00 00 04 03 05 06 05 04 05 07 05 03 04 05 // "NES C OMPLE"
03520: 06 03 04 05 04 04 00 03 00 00 04 09 04 06 00 00 06 02 06 06 // "TED. IF SW"
03540: 00 00 07 01 00 00 05 06 04 06 04 06 00 00 04 01 05 05 04 04 // " 1 OF F AND"
03560: 00 00 05 05 05 06 00 00 05 09 05 06 04 06 03 04 09 05 05 // " NO R OUTIN"
03580: 04 05 00 00 05 05 05 06 06 02 00 00 06 03 06 08 05 07 04 05 // "E NOS TYPE"
03600: 04 04 00 00 05 06 06 04 06 03 02 03 00 00 05 04 04 01 04 03 // "D OUT , MAC"
03620: 04 08 04 09 05 05 04 05 00 00 05 07 04 05 05 09 04 06 05 06 // "HINE PERFO"
03640: 05 09 05 04 04 05 04 04 00 00 06 03 04 05 06 02 06 03 06 02 // "RMED TESTS"
03660: 00 00 05 07 05 09 05 06 05 07 04 05 05 09 05 03 06 08 00 03 // " PROP ERLY."
03680: 00 00 00 0A 03 04 00 00 00 00 00 00 00 00 01 00 02 03 09 00 03 // " #@ . "
03700: 04 08 01 00 00 01 00 00 04 08 00 00 00 00 00 00 00 00 00 00 // "H+ H "
03720: 00 00 00 00 00 00 00 00 00 00 00 0A 00 00 00 00 00 00 00 // " # "
25840: 00 00 00 00 00 00 00 00 00 00 04 01 04 02 04 03 04 04 04 05 // " ABCDE"
25860: 04 06 04 07 04 08 04 09 05 01 05 02 05 03 05 04 05 05 05 06 // "FGHIJ KLMNO"
25880: 05 07 05 08 05 09 06 02 06 03 06 04 06 05 06 06 06 07 06 08 // "PQRST UVWXY"
25900: 06 09 00 03 00 04 01 00 01 03 01 04 02 00 02 01 02 03 02 04 // "Z.)+\$ *-/ , ("
25920: 03 03 03 04 07 00 07 01 07 02 07 03 07 04 07 05 07 06 07 07 // "=@012 34567"
25940: 07 08 07 09 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 // "89# "
39940: 00 00 00 00 00 00 00 00 00 00 00 00 03 08 03 09 09 09 04 00 // " "
39960: 00 01 00 00 04 09 00 03 02 02 08 0A 00 00 00 00 03 09 09 07 // " I. # "

39980: 06 00 01 02 03 04 05 06 07 08 09 0A 11 12 03 09 09 09 08 09 // " @08 # "
40000: 04 00 00 00 04 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 // " # "
45840: 00 00 00 00 00 00 00 00 00 00 07 00 07 01 07 02 07 03 07 04 // " 01234"
45860: 07 05 07 06 07 07 07 08 07 09 00 03 00 04 01 00 01 03 01 04 // "56789 .)+\$*"
45880: 02 00 02 01 02 03 02 04 03 03 03 04 04 01 04 02 04 03 04 04 // "-/, (= @ABCD"
45900: 04 05 04 06 04 07 04 08 04 09 05 01 05 02 05 03 05 04 05 05 // "EFGHI JKLMN"
45920: 05 06 05 07 05 08 05 09 06 02 06 03 06 04 06 05 06 06 06 07 // "OPQRS TUWX"
45940: 06 08 06 09 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 // "YZ# "
59940: 00 00 00 00 00 00 00 00 00 00 00 00 03 08 05 09 09 09 04 00 // " R "
59960: 00 01 00 00 04 09 00 03 04 02 00 0A 00 00 00 00 05 09 09 07 // " I.B # R "
59980: 06 00 01 02 03 04 05 06 07 08 09 0A 11 12 05 09 09 09 08 09 // " @08 # R "

// End

DX03– Typewriter Diagnostic


```
//=====
//  
// DX03 - Typewriter Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Bypass error type out  
// OFF - Type out routine number on error  
// PS2: ON - Loop in routine  
// OFF - Continue to next routine  
// PS3: ON - Stop on error  
// OFF - Do not stop on error, continue  
// PS4: ON - Repeat test DX03  
// OFF - Run test DX03 once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - STOP  
// O'FLOW - PROGRAM  
//  
// Start addresses:  
//  
// 00440 - Full test w/ title  
// 00726 - Full test w/o title  
// 01982 - Alphabetic echo test  
// 02256 - Numeric echo test  
//  
// Directions:  
//  
// 1. Load DX03 diagnostic  
// 2. Tab stops are automatically set at 45 & 75  
// 3. Press START  
// 4. Press START  
// 5. Press INSERT  
// 6. Type 4901982  
// 7. Press RELEASE  
// 8. Press START  
// 9. Type alphabetic characters, end with record mark  
// 10. Press RELEASE-START  
// 11. Press INSERT  
// 12. Type 4902256  
// 13. Press RELEASE  
// 14. Press START  
// 15. Type numeric characters, end with record mark  
// 16. Press RELEASE-START  
//=====
```


Sample Output - DX03

TYPEWRITER TEST DTX03

ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ

.)+\$*-/, (=@-Ø123456789
.)+\$*-/, (=@-Ø123456789
.)+\$*-/, (=@-Ø123456789

Ø123456789@
Ø123456789@
Ø123456789@

Ø123456789@‡
Ø123456789@‡
Ø123456789@‡

TABULATE	RETURN	CARRIAGE
TABULATE	RETURN	CARRIAGE
TABULATE	RETURN	CARRIAGE

6789	Ø12345
6789	Ø12345
6789	Ø12345

SPACE TWICE SPACE TWICE SPACE TWICE SPACE TWICE SPACE TWICE
DTX03 TYPEWRITER TEST COMPLETE

49Ø1982
KEY IN ALPHA RECORD, TERMINATE WITH RECORD MARK
ABCDEFGHIJKLMNOPQRSTUVWXYZ Ø123456789 @()=¤.-+\$/, .‡
ABCDEFGHIJKLMNOPQRSTUVWXYZ Ø123456789 @()=¤.-+\$/, .

49Ø2256
KEY IN NUMERIC RECORD, TERMINATE WITH RECORD MARK
Ø123456789Ø123456789‡
Ø123456789Ø123456789

NO. 2172336
SHEET 0
OF 12

DIAGNOSTIC TEST

DIPAL NAME DX03

TITLE TYPEWRITER TEST - DTX03
MACH. TYPE 1620 - I BY HNT APPR. DATE

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404980	5-7-64	1 - 12

E/C NO.	404980							
DATE	5-7-64							

DTX03
1620-1 TYPEWRITER TEST
DIPAL NAME DX03

PROGRAM DESCRIPTION

1. Purpose

This program tests all operation of the 1620-1 Console Typewriter, including read numeric and read alpha if desired.

2. Test Method

This program is a fault detection test and is made up of short, independent routines. Each operation and character set is executed three times before proceeding to the next routine Standard program switch control is provided.

3. Modifications

This program contains the necessary linkage to make it compatible with the DIPAL Monitor.

4. Units of System Required

This test requires the basic 1620-1 and 1622 card reader on 1621 paper tape reader.

5. E/C Level

All levels.

LOADING INSTRUCTIONS

1. Clear core by inserting 31 00003 00002.
Release/Start.
2. Load test deck from 1622 by pushing LOAD button.
3. Load from paper tape Insert 36 00000 00300 K-S
4. To reproduce the paper tape,
Insert 36 00000 00300, Release, Press
"SIE" Button. After tape has loaded insert

LOC
00000 35 00000 00200
00012 41 00416 00000
00024 16 00013 00049
00036 16 00001 00041
00048 49 00416 00000

4. Continued

Release/Start

When MAR reaches 03202, Stop by pressing "SCE" Button, Reset, and Insert 35 19999 00200. Release/Start

5. Machine will halt with 00011 in MAR. Press START to execute test.

OPERATING INSTRUCTIONS

1. The machine will type instructions for setting margins and tab stops. After setting these, press START to continue.
2. After one program pass is complete, pass complete message is typed out. If SW4 is on, the program is repeated. If SW4 is off, the next program is read from the 1622.
3. Read Alpha Test

A manual branch to location 01982 must be executed in order to enter the Read Alpha test routine. Instructions will be typed out, and then a Read Alpha instruction executed. Any length record may be entered. Terminate with a record mark, release and start. The same record will be typed out. Switch 2 will cause the routine to loop in either the read or write phase, and Switch 4 will repeat the complete Read Alpha test. If SW4 and SW2 are off, the program will halt after writing the record. Pressing START will cause the routine to repeat.

4. Read Numeric Test

Same as the Read Alpha test, except a manual branch to location 02256.

5. Program Console Switches

The program console switches have the following control:

Switch 1 ON: Bypass all error routines
 OFF: Test Switch 3 on error

Switch 2 ON: Loop in test routine
 OFF: Continue to next routine

Switch 3 ON: Halt on error if SW1 off
 OFF: Typeout on error if SW1 off

Switch 4 ON: Repeat program
 OFF: Load next program from 1622

6. Data Check Switches - as desired.

7. Normal Program Halts

00560 Allow Tab and Margin Settings
02138 Read Alpha Test Complete and SW4 OFF.
02412 Read Numeric Test Complete and SW4 OFF.

ERROR INFORMATION

1. Error Halts

There is one error halt in the program.

02710 Detected error. Display LR2 for location +12 of error exit.

2. Error Typeouts

Error messages are typed out in the following format:

ERROR EXIT ADR 01100 MBR-E CK MBR-O CK

TYPEWRITER TEST DTX03

Set Margins at 10 and 90, Tab stops at 55 and 85

ABCDEFGHIJKLMNPQRS TUWXYZ
ABCDEFGHIJKLMNPQRS TUWXYZ
ABCDEFGHIJKLMNPQRS TUWXYZ

.)+\$*-/, (=@-0123456789
.)+\$*-/, (=@-0123456789
.)+\$*-/, (=@-0123456789

0123456789@
0123456789@
0123456789@

0123456789@#
0123456789@#
0123456789@#

TABULATE RETURN CARRIAGE
TABULATE RETURN CARRIAGE
TABULATE RETURN CARRIAGE

6789

6789

6789

DTX03 Page 3
PN 2172336
EC 404980

SPACE TWICE SPACE TWICE SPACE TWICE SPACE TWICE SPACE TWICE
DTX03 TYPEWRITER TEST COMPLETE

4901982RS

KEY IN ALPHA RECORD, TERMINATE WITH RECORD MARK
ABCDEFGHIJKLMNOPQRSTUVWXYZU10JKLM,.+\$-/@()#RS
ABCDEFGHIJKLMNOPQRSTUVWXYZU10JKLM,.+\$-/@()

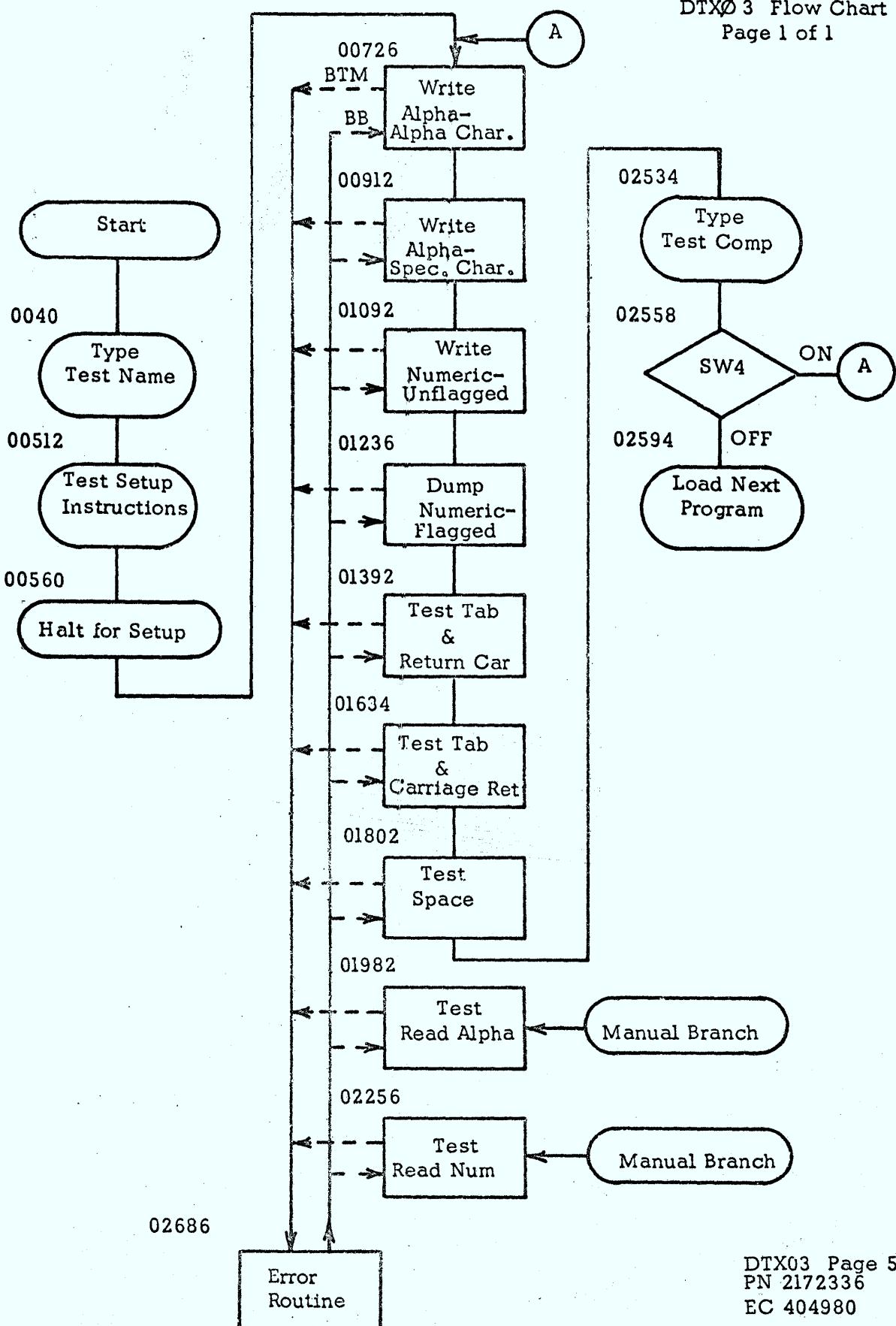
4902256RS

KEY IN NUMERIC RECORD, TERMINATE WITH RECORD MARK

12345678912345678900#RS

12345678912345678900@

DTX03 Page 4
PN 2172336
EC 404980



* DTX03
 * 1620-1 TYPEWRITER TEST
 *

00402		DORG 402
18000	00000	MONIT DS ,18000
18117	00000	MON DS ,18117
00403	00004	NAME DAC 4,DX03,
00414	00005 -3202	DSA LAST
00416 45	00440 18117	ORG BNR *+24,MON
00428 49	18000 00000	B MONIT
00440 34	00726 00102	START RCTY T1.0
00452 16	00441 000M9	TFM *-11,49,10
00464 39	00585 00100	WATY IDEN,,, TYPE TEST NAME
00476 47	00500 01900	BNA *+24,,, TEST ANY DATA CK
00488 17	02686 -0488	BTM ERROR,,, GO TO ERROR ROUTINE
00500 34	00000 00102	RCTY
00512 39	00629 00100	WATY INST,,, TYPE INSTRUCTIONS
00524 47	00548 01900	BNA *+24,,, TEST ANY DATA CK
00536 17	02686 -0536	BTM ERROR,,, GO TO ERROR ROUTINE
00548 34	00000 00102	RCTY
00560 48	00000 00000	H ,,, HALT TO ALLOW TAB SETTING
00572 49	00726 00000	B T1.0,,, GO TO FIRST TEST
00585	00022	IDEN DAC 22,TYPEWRITER TEST DTX03*
00629	00025	INST DAC 25,SET MARGINS AT 10 AND 90,
00679	00023	DAC 23,TAB STOPS AT 55 AND 85*

*
 * THIS ROUTINE WILL TYPE THE ALPHABETIC
 * CHARACTERS 3 TIMES.
 *

00725	00002	COUNT DC 2,0,, 2 DIGIT FIELD FOR PASS COUNT
00726 16	00725 000-0	T1.0 TFM COUNT,0,10, RESET PASS COUNT TO 0
00738 34	00000 00102	RCTY
00750 34	00000 00102	RCTY
00762 39	00859 00100	WATY K1.0,,, TYPE FIRST PATTERN
00774 47	00798 01900	BNA *+24,,, TEST ANY DATA CK
00786 17	02686 -0786	BTM ERROR,,, GO TO ERROR ROUTINE
00798 46	00750 00200	BC2 T1.0+24,,, LOOP IF SW2 ON
00810 11	00725 000-1	AM COUNT,1,10, UPDATE PASS COUNT
00822 14	00725 000-3	CM COUNT,3,10, SEE IF 3 LINES TYPED
00834 47	00750 01300	BL T1.0+24,,, GO TYPE ANOTHER LINE
00846 49	00912 00000	B T2.0,,, GO TO NEXT TEST

00859	00027	K1.0 DAC 27,ABCDEFIGHJKLMNOPQRSTUVWXYZ*
-------	-------	---

*
 * THIS ROUTINE WRITES ALL NUMERALS AND
 * SPECIAL CHARACTERS 3 TIMES IN ALPHA MODE.
 *

00912 16	00725 000-0	T2.0 TFM COUNT,0,10, RESET PASS COUNT TO 0
00924 34	00000 00102	RCTY
00936 34	00000 00102	RCTY
00948 39	01045 00100	WATY K2.0,,, TYPE TEST PATTERN
00960 47	00984 01900	BNA *+24,,, TEST ANY DATA CK
00972 17	02686 -0972	BTM ERROR,,, GO TO ERROR ROUTINE
00984 46	00936 00200	BC2 T2.0+24,,, LOOP IF SW2 ON
00996 11	00725 000-1	AM COUNT,1,10, UPDATE PASS COUNT

01008 14 00725 000-3	CM COUNT,3,10, SEE IF 3 LINES TYPED
01020 47 00936 01300	BL T2.0+24,,, GO TYPE ANOTHER LINE
01032 49 01092 00000	B T3.0,,, GO TO NEXT TEST
*	
01045 00024	K2.0 DAC 24, .)+\$*-/, (='-0123456789'
*	
* THIS ROUTINE WILL TYPE 3 LINES OF UNFLAGGED NUMERIC CHARACTERS.	
*	
01092 16 00725 000-0	T3.0 TFM COUNT,0,10, RESET PASS COUNT TO 0
01104 34 00000 00102	RCTY
01116 34 00000 00102	RCTY
01128 38 01224 00100	WNTY K3.0,,, TYPE TEST PATTERN
01140 47 01164 01900	BNA *+24,,, TEST ANY DATA CK
01152 17 02686 -1152	BTM ERROR,,, GO TO ERROR ROUTINE
01164 46 01116 00200	BC2 T3.0+24,,, LOOP IF SW2 ON
01176 11 00725 000-1	AM COUNT,1,10, UP DATE PASS COUNT
01188 14 00725 000-3	CM COUNT,3,10, SEE IF 3 LINES TYPED
01200 47 01116 01300	BL T3.0+24,,, GO TYPE ANOTHER LINE
01212 49 01236 00000	B T4.0,,, GO TO NEXT TEST
*	
01224 00010	K3.0 DSC 10,0123456789
01234 00001	DNB 1,,, 8-4 CHAR
01235 00001	DC 1,'', RECORD MARK
*	
* THIS ROUTINE TYPES 3 LINES OF FLAGGED NUMERIC CHARACTERS USING DUMP NUMERIC INSTRUCTION.	
*	
01236 16 00725 000-0	T4.0 TFM COUNT,0,10, RESET PASS COUNT TO 0
01248 31 19988 01380	TR 19988,K4.0,,, SET UP DATA FIELD
01260 34 00000 00102	RCTY
01272 34 00000 00102	RCTY
01284 35 19988 00100	DNTY 19988,,, TYPE TEST PATTERN
01296 47 01320 01900	BNA *+24,,, TEST ANY DATA CK
01308 17 02686 -1308	BTM ERROR,,, GO TO ERROR ROUTINE
01320 46 01272 00200	BC2 T4.0+36,,, LOOP IF SW2 ON
01332 11 00725 000-1	AM COUNT,1,10, UPDATE PASS COUNT
01344 14 00725 000-3	CM COUNT,3,10, SEE IF 3 LINES TYPED
01356 47 01272 01300	BL T4.0+36,,, GO TYPE ANOTHER LINE
01368 49 01392 00000	B T5.0,,, GO TO NEXT TEST
*	
01380 00001	K4.0 DC 1,-0
01382 00002	DC 2,-12
01384 00002	DC 2,-34
01386 00002	DC 2,-56
01388 00002	DC 2,-78
01389 00001	DC 1,-9
01390 00001	DNB 1,,, 8-4 CHAR
01391 00001	DC 1,'', RECORD MARK
*	
* THIS ROUTINE TESTS THE TABULATE AND RETURN CARRIAGE OPERATIONS.	
*	
01392 16 00725 000-0	T5.0 TFM COUNT,0,10, RESET PASS COUNT TO ZERO
01404 34 00000 00102	RCTY
01416 34 00000 00102	RCTY
01428 39 01585 00100	WATY K5.0,,, TYPE TABULATE
01440 34 00000 00108	TBTY ,,, TAB TO 45

01452	47	01476	01900	BNA	*+24,,,	TEST ANY DATA CK
01464	17	02686	-1464	BTM	ERROR,,,	GO TO ERROR ROUTINE
01476	39	01603	00100	WATY	K5.1,,,	TYPE RETURN CARRIAGE
01488	34	00000	00102	RCTY		
01500	47	01524	01900	BNA	*+24,,,	TEST ANY DATA CK
01512	17	02686	-1512	BTM	ERROR,,,	GO TO ERROR ROUTINE
01524	46	01428	00200	BC2	T5.0+36,,,	LOOP IF SW2 ON
01536	11	00725	000-1	AM	COUNT,1,10,	UPDATE PASS COUNT
01548	14	00725	000-3	CM	COUNT,3,10,	SEE IF 3 PASSES MADE
01560	47	01428	01300	BL	T5.0+36,,,	DO AGAIN
01572	49	01634	00000	B	T6.0,,,	GO TO NEXT TEST
*						
01585		00009		K5.0	DAC	9,TABULATE*
01603		00016		K5.1	DAC	16,RETURN CARRIAGE*
*						
*						
THIS ROUTINE TESTS THE TABULATE AND						
CARRIAGE RETURN.						
*						
01634	16	00725	000-0	T6.0	TFM	COUNT,0,10, RESET PASS COUNT TO 0
01646	34	00000	00102		RCTY	
01658	34	00000	00102		RCTY	
01670	34	00000	00108		TBTY	,,, TAB CARRIAGE
01682	34	00000	00108		TBTY	,,, TAB CARRIAGE
01694	38	01790	00100		WNTY	K6.0,,, TYPE TEST PATTERN
01706	47	01730	01900		BNA	*+24,,, TEST ANY DATA CK
01718	17	02686	-1718		BTM	ERROR,,, GO TO ERROR ROUTINE
01730	46	01658	00200		BC2	T6.0+24,,, LOOP IF SW? ON
01742	11	00725	000-1		AM	COUNT,1,10, UPDATE PASS COUNT
01754	14	00725	000-3		CM	COUNT,3,10, SEE IF 3 PASSES MADE
01766	47	01658	01300		BL	T6.0+24,,, DO AGAIN
01778	49	01802	00000		B	T7.0,,, GO TO NEXT TEST
*						
01790		00011		K6.0	DSC	11,0123456789*
*						
*						
THIS ROUTINE TESTS THE SPACE OPERATION						
*						
01802	16	00725	000-0	T7.0	TFM	COUNT,0,10, RESET PASS COUNT TO 0
01814	34	00000	00102		RCTY	
01826	34	00000	00102		RCTY	
01838	39	01959	00100		WATY	K7.0,,, TYPE-SPACE TWICE
01850	34	00000	00101		SPTY	
01862	34	00000	00101		SPTY	
01874	47	01898	01900		BNA	*+24,,, TEST ANY DATA CHECK
01886	17	02686	-1886		BTM	ERROR,,, GO TO ERROR ROUTINE
01898	46	01838	00200		BC2	T7.0+36,,, LOOP IF SW2 ON
01910	11	00725	000-1		AM	COUNT,1,10, UPDATE PASS COUNT
01922	14	00725	000-5		CM	COUNT,5,10, SEE IF 5 LOOPS MADE
01934	47	01838	01300		BL	T7.0+36,,, REPEAT
01946	49	02534	00000		B	FINISH,,, GO TO PROG COMPLETE ROUTINE
*						
01959		00012		K7.0	DAC	12,SPACE TWICE*
*						
*						
A MANUAL ENTRY TO THIS ROUTINE WILL SET UP						
A READ ALPHA OPERATION, AND WILL ACCEPT ANY						
ALPHA RECORD KEYED IN. THE SAME MESSAGE WILL						
BE TYPED OUT						

01982 34 00000 00102	T8.0	RCTY WATY INST2,,, RCTY RATY RDIN,,, BNA *+24,,, BTM ERROR,,, BC2 T8.0,,, RCTY WATY RDIN,,, BNA *+24,,, BTM ERROR,,, BC2 *-48,,, BC4 T8.0,,, H B T8.0,,,	TYPE INSTRUCTIONS READ IN RECORD TEST ANY DATA CK GO TO ERROR ROUTINE LOOP IN READ SECTION TYPE SAME RECORD OUT TEST ANY DATA CK GO TO ERROR ROUTINE LOOP IN WRITE SECTION LOOP IN ROUTINE IF SW4 ON REPEAT ROUTINE
01994 39 02163 00100			
02006 34 00000 00102			
02018 37 03199 00100			
02030 47 02054 01900			
02042 17 02686 -2042			
02054 46 01982 00200			
02066 34 00000 00102			
02078 39 03199 00100			
02090 47 02114 01900			
02102 17 02686 -2102			
02114 46 02066 00200			
02126 46 01982 00400			
02138 48 00000 00000			
02150 49 01982 00000			
02163 00047	INST2	DAC 47,KEY IN ALPHA RECORD,TERMINATE WITH RECORD MARK*	
	*		
	*	A MANUAL ENTRY TO THIS ROUTINE WILL SET UP	
	*	A READ NUMERIC OPERATION, AND WILL ACCEPT	
	*	ANY NUMERIC RECORD KEYED IN. THE SAME MESSAGE	
	*	WILL BE TYPED OUT.	
	*		
02256 34 00000 00102	T9.0	RCTY WATY INST3,,, RCTY RNRY RDIN,,, DNA *+24,,, BTM ERROR,,, BC2 T9.0,,, RCTY WNTY RDIN,,, BNA *+24,,, BTM ERROR,,, BC2 *-48,,, BC4 T9.0,,, H B T9.0,,,	TYPE INSTRUCTIONS READ IN RECORD TEST ANY DATA CK GO TO ERROR ROUTINE LOOP IN READ SECTION TYPE SAME RECORD OUT TEST ANY DATA CK GO TO ERROR ROUTINE LOOP IN WRITE SECTION LOOP IN ROUTINE REPEAT ROUTINE
02268 39 02437 00100			
02280 34 00000 00102			
02292 36 03199 00100			
02304 47 02328 01900			
02316 17 02686 -2316			
02328 46 02256 00200			
02340 34 00000 00102			
02352 38 03199 00100			
02364 47 02388 01900			
02376 17 02686 -2376			
02388 46 02340 00200			
02400 46 02256 00400			
02412 48 00000 00000			
02424 49 02256 00000			
02437 00049	INST3	DAC 49,KEY IN NUMERIC RECORD,TERMINATE WITH RECORD MARK*	
	*		
	*	PROGRAM FINISHED ROUTINE	
02534 34 00000 00102	FINISH	RCTY WATY FINI BC4 T1.0,,, BNR *+24,MON B MONIT RNCD,,, B 0,,, FINI DAC 31,DTX03 TYPEWRITER TEST COMPLETE*	REPEAT PROG IF SW4 ON READ NEXT PROG IF SW4 OFF GO TO LOADER
02546 39 02619 00100			
02558 46 00726 00400			
02570 45 02594 18117			
02582 49 18000 00000			
02594 36 00000 00500			
02606 49 00000 00000			
02619 00031			
02684 00005			
	*		
	*	COMMON ERROR ROUTINE	
02686 46 02938 00100	ERROR	BC1 RSET,,, BNC3 *+36,,,	BYPASS ERROR RTN IF SW1 ON BYPASS HALT IF SW3 OFF
02698 47 02734 00300			

02710	48	00000	00000	H ,,,	DISPLAY IR2 FOR LOC+12 OF ERROR EXIT
02722	49	02938	00000	B RSET,,,	GO RESET DATA CK INDICATORS
*				ERROR TYPEOUT SECTION	
02734	26	03060	02685	TF OUT+4,ERROR-1,,SET EXIT ADR IN TYPEOUT AREA	
02746	34	00000	00102	RCTY	
02758	39	03011	00100	WATY EXT,,,	
02770	38	03056	00100	WNTY OUT,,,	TYPE ADDRESS
*				TYPE OUT WHICH DATA CK INDICATORS ARE ON	
02782	47	02806	00600	BNI #+24,600,,	READ CK
02794	39	03043	00100	WATY RDK	
02806	47	02830	00700	BNI #+24,700,,	WRITE CK
02818	39	03063	00100	WATY WRK	
02830	47	02854	00800	BNI #+24,800,,	MAR CK
02842	39	03077	00100	WATY MRK	
02854	47	02878	01600	BNI #+24,1600,,	MBRE CK
02866	39	03093	00100	WATY BRE	
02878	47	02902	01700	BNI #+24,1700,,	MBRO CK
02890	39	03113	00100	WATY BRO	
02902	47	02926	01900	BNI #+24,1900,,	ANY DATA CK
02914	39	03133	00100	WATY HUH	
02926	42	00000	00000	BB ,,,	RETURN TO PROG
*				IND RESET IF SW1 OR 3 ON	
02938	46	02950	00600	RSET BI #+12,600	
02950	46	02962	00700	BI #+12,700	
02962	46	02974	00800	BI #+12,800	
02974	46	02986	01600	BI #+12,1600	
02986	46	02998	01700	BI #+12,1700	
02998	42	00000	00000	BB ,,,	RETURN TO PROG
03011	00016	EXT	DAC	16,ERROR EXIT ADR	*
03043	00007	RDK	DAC	7, RD CK	*
03056	00006	OUT	DSC	6,0	*
03063	00007	WRK	DAC	7, WR CK	*
03077	00008	MRK	DAC	8, MAR CK	*
03093	00010	BRE	DAC	10, MBR-E CK	*
03113	00010	BRO	DAC	10, MBR-O CK	*
03133	00033	HUH	DAC	33, DATA CK CAUSING ERROR NOT RESET	*
03199	00002	RDIN	DAC	2,	
03202	00001	LAST	DSC	1,0	
00416			DEND	ORG	

DT X03 80/80 LIST

360007200500360020100500440001200276260005900274250001100000260009000269 -0000
 26000950026431000000020026001140027425000000011490001200000 -0001
 M4677073# 1-1-0402-0410 -0002
 -3202# 1-1-0410-0415 -0003
 4500440181174918000000003400726001021600441000M9390058500100#0-1-0416-0476 -0004
 4700500019001702686-048834000000102390062900100470054801900#0-1-0476-0536 -0005
 1702686-053634000000102480000000049007260000#0-1-0536-0584 -0006
 0368574566594963455900634562630044636770730# 1-1-0584-0628 -0007
 0245630054415947495562004163007170041554400797023# 1-1-0628-0678 -0008
 034142006263565762004163007575004155440078750# 1-1-0678-0724 -0009
 -0# 1-1-0724-0726 -0010
 1600725000-0340000000102340000000102390085900100470079801900#0-1-0726-0786 -0011
 1702686-07864600750002001100725000-11400725000-3470075001300#0-1-0786-0846 -0012
 490091200000# 0-1-0846-0858 -0013
 M1424344454647484951525354555657585962636465666768690# 1-1-0858-0912 -0014
 1600725000-0340000000102340000000102390104500100470098401900#0-1-0912-0972 -0015
 1702686-0972460093602001100725000-11400725000-3470093601300#0-1-0972-1032 -0016
 490109200000# 0-1-1032-1044 -0017
 -0030410131420212324333450707172737475767778790# 1-1-1044-1092 -0018
 1600725000-0340000000102340000000102380122400100470116401900#0-1-1092-1152 -0019
 1702686-11524601116002001100725000-11400725000-3470111601300#0-1-1152-1212 -0020
 490123600000# 0-1-1212-1224 -0021
 0123456789# 1-1-1224-1234 -0022
 *# -7-1234-1235 -0023
 *# 1-1-1235-1236 -0024
 *#
 1600725000-0311996801380340000000102340000000102351998800100#0-1-1236-1296 -0025
 4701320019001702686-13084601272002001100725000-11400725000-3#0-1-1296-1356 -0026
 47012720130049013920000# 0-1-1356-1380 -0027
 -JKLMNOPQR# 1-1-1380-1390 -0028
 *# -7-1390-1391 -0029
 *# 1-1-1391-1392 -0030
 1600725000-0340000000102340000000102390158500100340000000108#0-1-1392-1452 -0031
 4701476019001702686-1464390160300100340000000102470152401900#0-1-1452-1512 -0032
 1702686-15124601428002001100725000-11400725000-3470142801300#0-1-1512-1572 -0033
 490163400000# 0-1-1572-1584 -0034
 03414264534163450# 1-1-1584-1602 -0035
 N945636459550043415959494147450# 1-1-1602-1634 -0036
 1600725000-0340000000102340000000102340000000108340000000108#0-1-1634-1694 -0037
 3801790001004701730019C01702686-17184601658002001100725000-1#0-1-1694-1754 -0038
 1400725000-34701658013C0490180200000# 0-1-1754-1790 -0039
 0123456789# 1-1-1790-1801 -0040
 1600725000-034000000102340000000102390195900100340000000101#0-1-1802-1862 -0041
 3400000001014701898019001702686-18864601838002001100725000-1#0-1-1862-1922 -0042
 1400725000-5470183801300490253400000# 0-1-1922-1958 -0043
 02574143450063664943450# 1-1-1958-1982 -0044
 3400000001023902163001C0340000000102370319900100470205401900#0-1-1982-2042 -0045
 1702686-20424601982002C0340000000102390319900100470211401900#0-1-2042-2102 -0046
 1702686-21024602066002C0460198200400480000000004#0198200000#0-1-2102-2162 -0047
 N24568004955004153574841005945435659442363455944# 1-1-2162-2212 -0048
 5541634500664963480059454356594400544159520# 1-1-2212-2256 -0049
 3400000001023902437001C0340000000102360319900100470232801900#0-1-2256-2316 -0050
 1702686-2316460225600200340000000102380319900100470238801900#0-1-2316-2376 -0051
 1702686-2376460234000200460225600400480000000004#0225600000#0-1-2376-2436 -0052
 N2456800495500556454455949430059454356594423634559# 1-1-2436-2486 -0053
 54495541634500664963480059454356594400544159520# 1-1-2486-2534 -0054
 340000000102390261900100460072600400450259418117491800000000#0-1-2534-2594 -0055
 36000000050049000000000# 0-1-2594-2618 -0056
 M4636770730063685745665949634559006345626300435654# 1-1-2618-2668 -0057

57534563450*	1-1-2668-2680	-0058
-0000*	1-1-2680-2685	-0059
460293800100470273400300480000000000490293800000260306002685*	0-1-2686-2746	-0060
340000000102390301100100380305600100470280600600390304300100*	0-1-2746-2806	-0061
470283000700390306300100470285400800390307700100470287801600*	0-1-2806-2866	-0062
390309300100470290201700390311300100470292601900390313300100*	0-1-2866-2926	-0063
420000000000460295000600460296200700460297400800460298601600*	0-1-2926-2986	-0064
460299801700420000000000*	0-1-2986-3010	-0065
M559595659004567496300414459000*	1-1-3010-3042	-0066
-059440043520*	1-1-3042-3056	-0067
00000*	1-1-3056-3062	-0068
-066590043520*	1-1-3062-3076	-0069
-05441590043520*	1-1-3076-3092	-0070
-054425920450043520*	1-1-3092-3112	-0071
-054425920700043520*	1-1-3112-3132	-0072
-0444163410043520043416462495547004559595659005556*	1-1-3132-3182	-0073
630059456245630*	1-1-3182-3198	-0074
-00*	1-1-3198-3202	-0075
0*	1-1-3202-3203	-0076
00000 L600000005004900000*1205723-000133057230000049057120000-8-0096-0115	-0077	
360010000500360017200500360024400500360031600500360000000500	-0078	
00000000000102030400020406080003060902100408021610050015102006021814200*	-0079	
704112820080614223009081726300000000005060708090012141618151811242720242*	-0080	
822363520353045403632484455324946536048465462754453627180123456789123456*	-0081	
789-23456789-J3456789-JK456789-JKL56789-JKLM6789-JKLMN789-JKLMN089-JKLMN*	-0082	
M8000000000049-04160P9-JKLMNOPQ*	L10038800019M90000000000M90003600000	-0083

IO02– Card I/O Diagnostic


```
//=====
//  
// IO02 - Card I/O Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Bypass error type out  
// OFF - Type out routine number on error  
// PS2: ON - Loop in routine  
// OFF - Continue to next routine  
// PS3: ON - Stop on error  
// OFF - Do not stop on error, continue  
// PS4: ON - Repeat test IO02  
// OFF - Run test IO02 once  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - PROGRAM  
// O'FLOW - PROGRAM  
//  
// Start addresses:  
//  
// 00828 - Full test  
// 07152 - Punch alphabetic ripple deck  
// 02796 - Verify alphabetic ripple deck  
// 07188 - Punch numeric ripple deck  
// 05304 - Verify numeric ripple deck  
//  
// Directions:  
//  
// 1. Load IO02 diagnostic  
// 2. Press START  
// 3. Press RELEASE  
// 4. Press RESET  
// 5. Press INSERT  
// 6. Type 4907152  
// 7. Press RELEASE-START  
// 8. Verify that PUNCH NO FEED light is lit  
// 9. Insert empty card deck into punch  
// 10. Press START  
// 11. Verify that READER NO FEED light is lit  
// 12. Move card deck from punch to reader  
// 13. Press RELEASE  
// 14. Press RESET  
// 15. Press INSERT  
// 16. Type 4907188  
// 17. Press RELEASE-START  
// 18. Verify that PUNCH NO FEED light is lit  
// 19. Insert empty card deck into punch  
// 20. Press START  
// 21. Verify that READER NO FEED light is lit  
// 22. Move card deck from punch to reader  
//=====
```


Sample Output - IO02

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR IØ02 THEN START.
KEY IN 8Ø ALPHA CHARACTERS TO GENERATE RIPPC E DECK.

49Ø7152S

RIPPLE DECK PUNCHED LOAD INTO READER - START.

ALPHA OK - KEY IN 8Ø NUMERIC CHARACTERS TO GENERATE RIPPLE DECK.

49Ø7188S

RIPPLE DECK PUNCHED LOAD INTO READER - START.

IF NO ETOS IØ02 SUCCESSFUL.

NO. 2125684
SHEET 0
OF 27

DIAGNOSTIC TEST

TITLE 1622 CARD INPUT/OUTPUT DIAGNOSTIC TEST - IO02
MACH. TYPE 1620 BY J.H.M. APPR. G.I.A. DATE 4-11-62

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404618	5-15-61	1-26
404675	4-11-62	2, 3, 4, 10, 12, 13, 15, 19, 20 22, 23, 24, 25, 26, 27
404839	2-27-63	5, 8

E/C NO.	404618	404675	404839				
DATE	5-15-61	4-11-62	2-27-63				

1622 DIAGNOSTICS

TEST J002

A. SCOPE:

This test was designed to check all characters in all positions of the buffers in the 1622 punch and read, the last card indicator, and the read and write checks; both in alpha and numeric modes.

B. SET UP:

Sense Switch settings give the same result as standard 1620 diagnostic programs. To obtain the greatest amount of printed information in regard to errors place all Sense Switches in the Off position.

C. TEST METHOD:

The first two cards of the test deck contain loading instructions for the math tables which are contained on the following five cards. The eighth card contains the loading instructions for the program.

The entire deck plus two blank cards at the end are loaded into the reader bed, and with the computer in manual mode the Load Key is depressed.

When the last card has been read, a printout advising of the Sense Switch settings occurs and the computer halts.

Depressing the computer Start key continues the program until instructions are printed out to key in 80 alpha characters. Load the punch bed with a deck of blank cards and depress the card reader Start key on the punch. At this point the operator has a choice of one of two options:

- I. Key in 80 characters, release, start:
- II. Reset, insert, key in: 4907152 Release, Start.
(This causes a ripple deck of 80 cards to be punched.)

Option I allows the operator to select any desired sequence of characters, or any characters except the record mark he may wish including spacings. One need not key in 80 characters. A count of keyed in characters is unnecessary because an automatic function of the program allows only the first 80 characters to be used.

Option II allows the operator to use a table of 80 characters previously placed in core storage by the program. Option II is selected by branching to 07152 for alphabetic and to 07188 for numeric.

Option I may be preferable when a trouble is known to exist in a definite sequence.

A write check error occurring during the punching of the ripple deck causes a print out advising of such and a halt. Depression of the computer Start key reinitiates conditions for a restart without the need to reload the program.

When the printout occurs advising completion of the punching, depress Non Process Run out keys on the reader and punch. Remove the last two cards, (they will be unpunched), from the deck and place the deck in the read bed. Removal of last two cards is necessary to check the Last Card Indicator. Depress the reader Start key. Depress computer Start key.

C. TEST METHOD - continued

When the ripple deck is being read a noticeable variation of the reading rate of the cards may be observed. This being due to the searching of the computer through a comparison of a table. The rate may be fluctuated by shuffling the ripple deck prior to placing it in the read bed.

The program provides up to three attempts to successfully transfer the information read from the card from the 1622 to the 1620 in the event of a Read Check on the first attempt. If a Read Check occurs on the first transfer but the second or third attempted transfer is successful, a typeout stating the specific circumstance will occur. If a Read Check occurs on all three attempts, a print out advising of a restart is made. A restart at this point requires reloading of the ripple deck in the read bed.

If a card does not compare correctly after being read and all Sense Switches are off, a branch to error routine H001 is performed. After printing out H001 and the typewriter carriage is returned, a printout of the information contained on the erroneous card is performed followed by a print out of the table with which a comparison was attempted.

The table was generated automatically by the keyed in 80 characters doubled. By visually checking the card erroneously read, the operator can ascertain whether the punch incorrectly punched the data or the reader incorrectly read the punched data. Because each character keyed in passes through the punch and read buffers at least one time a defective punch or read circuit may be located.

Depressing computer Start Key continues the reading of the ripple deck with the card following the erroneous card.

An excellent check of the error portion of this program would be to insert a blank card or a card with known data into the ripple deck prior to loading into the reader and observing the results.

C. TEST METHOD - continued

The reader stops after the 78th card is read. It is necessary to depress the reader Start key to read the final two cards and check the Last Card Indicator.

Two checks of the Last Card Indicator are performed. The first checks that it is on and the second checks that it is turned off after inquiry.

H002 - Last Card indicator not on.

H003 - Last Card indicator not off.

The numeric portion of the test is conducted in the same manner as the alpha.

Option II will be 49 07188

H004 will be analogous to H001

H005 will be analogous to H002

H006 will be analogous to H003.

The ripple deck reading portions of the test may be run individually using prepunched ripple decks by branching to 02796 for alphabetic and to 05304 for numeric.

The complete normal typeout information will be as follows: (Note: The specific alphabetic and numeric data keyed in is optional to the operator; or options I and/or II may be used.)

SW 1 OFF SW 2 OFF SW 3 OFF SW 4 OFF SET SWS FOR 1002
THEN START.

KEY IN 80 ALPHA CHARACTERS TO GENERATE RIPPLE DECK.
ABCDEFGHIJKLMNPQRSTUVWXYZ0123456789@()=*.+-\$/.,

ABCDEFGHIJKLMNPQRSTUVWXYZ0123456

RIPPLE DECK PUNCHED LOAD INTO READER - START.

ALPHA OK - KEY IN 80 NUMERIC CHARACTERS TO GENERATE
RIPPLE DECK.

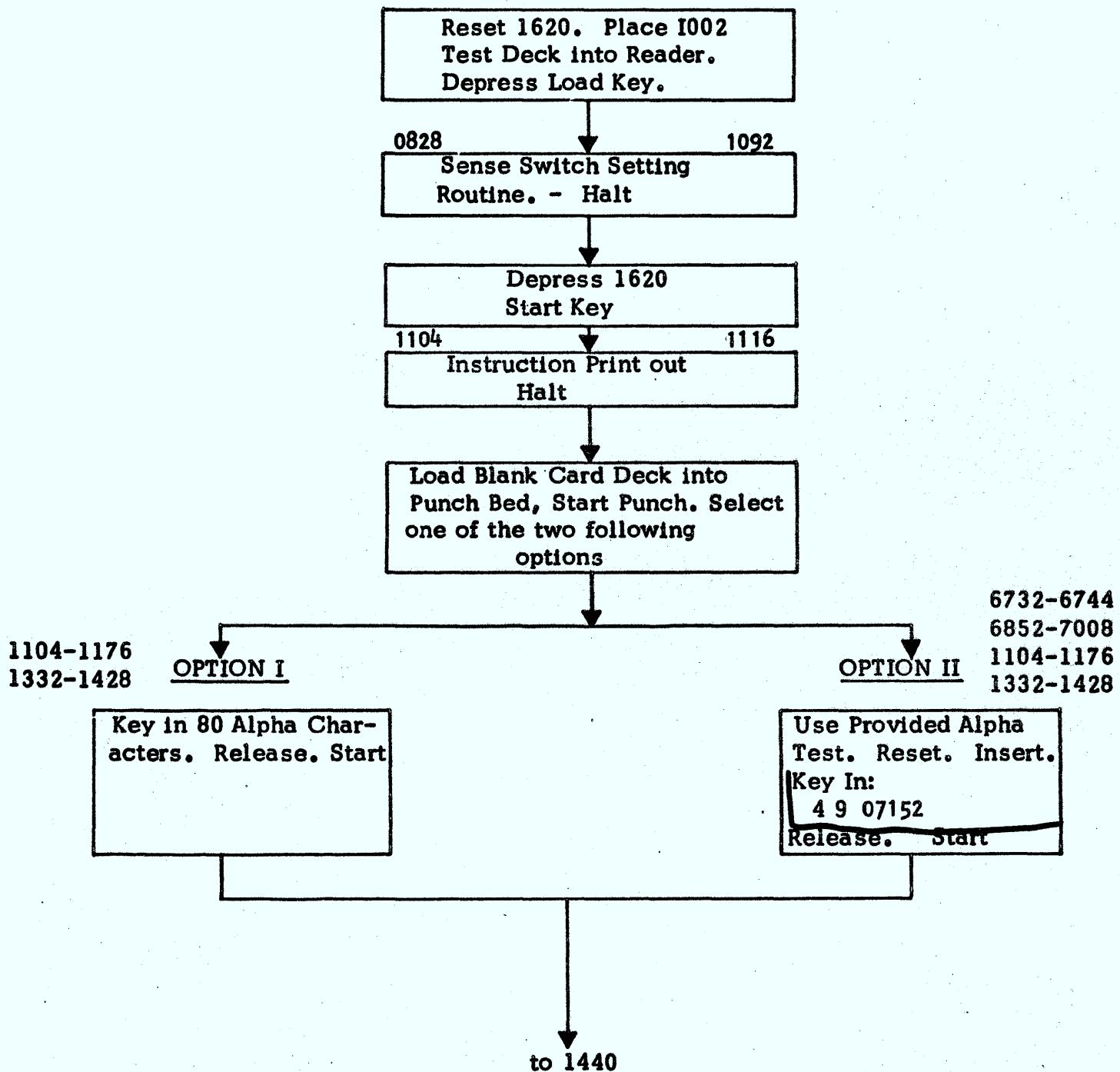
0123456789012345678901234567890123456789

0123456789012345678901234567890123456789

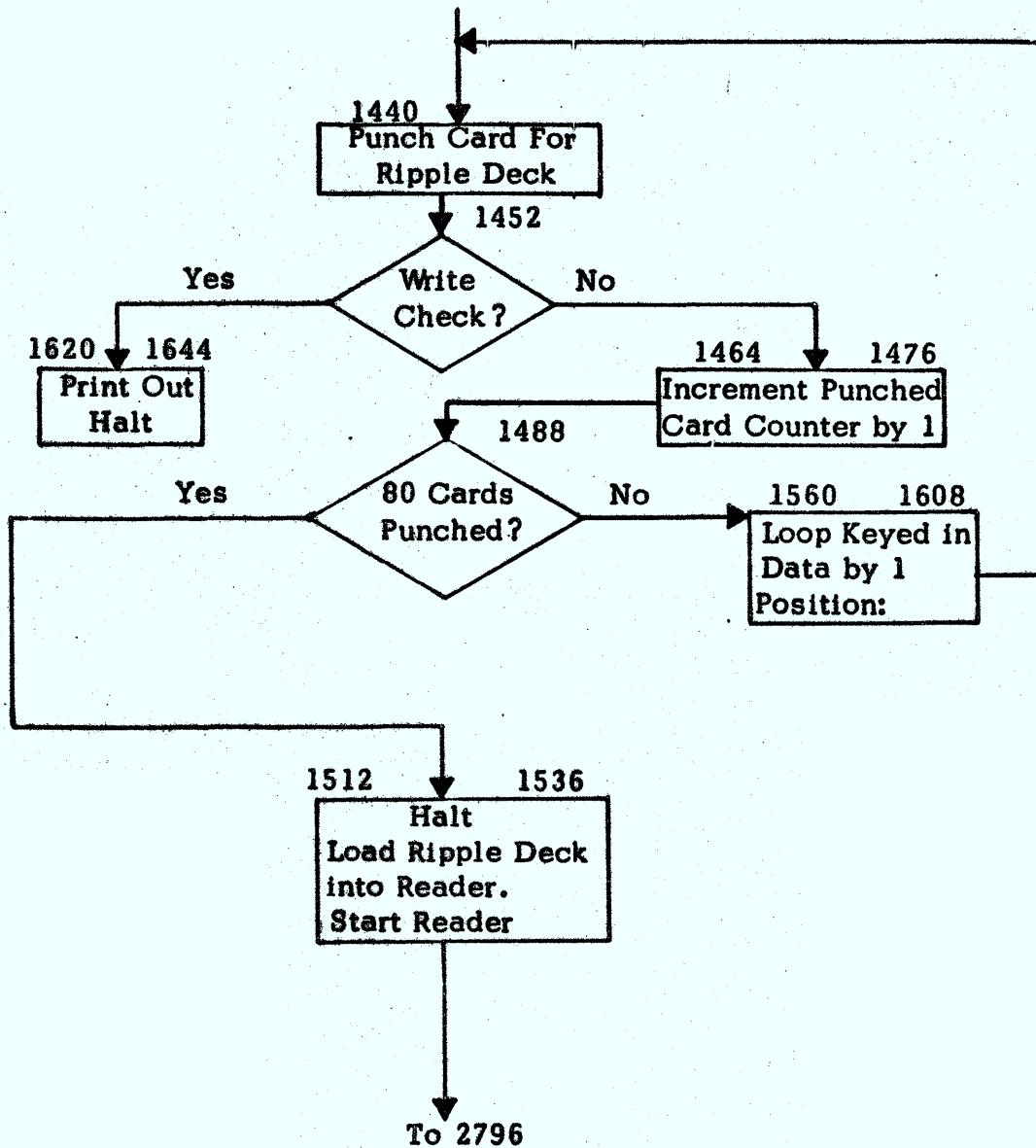
RIPPLE DECK PUNCHED LOAD INTO READER - START.

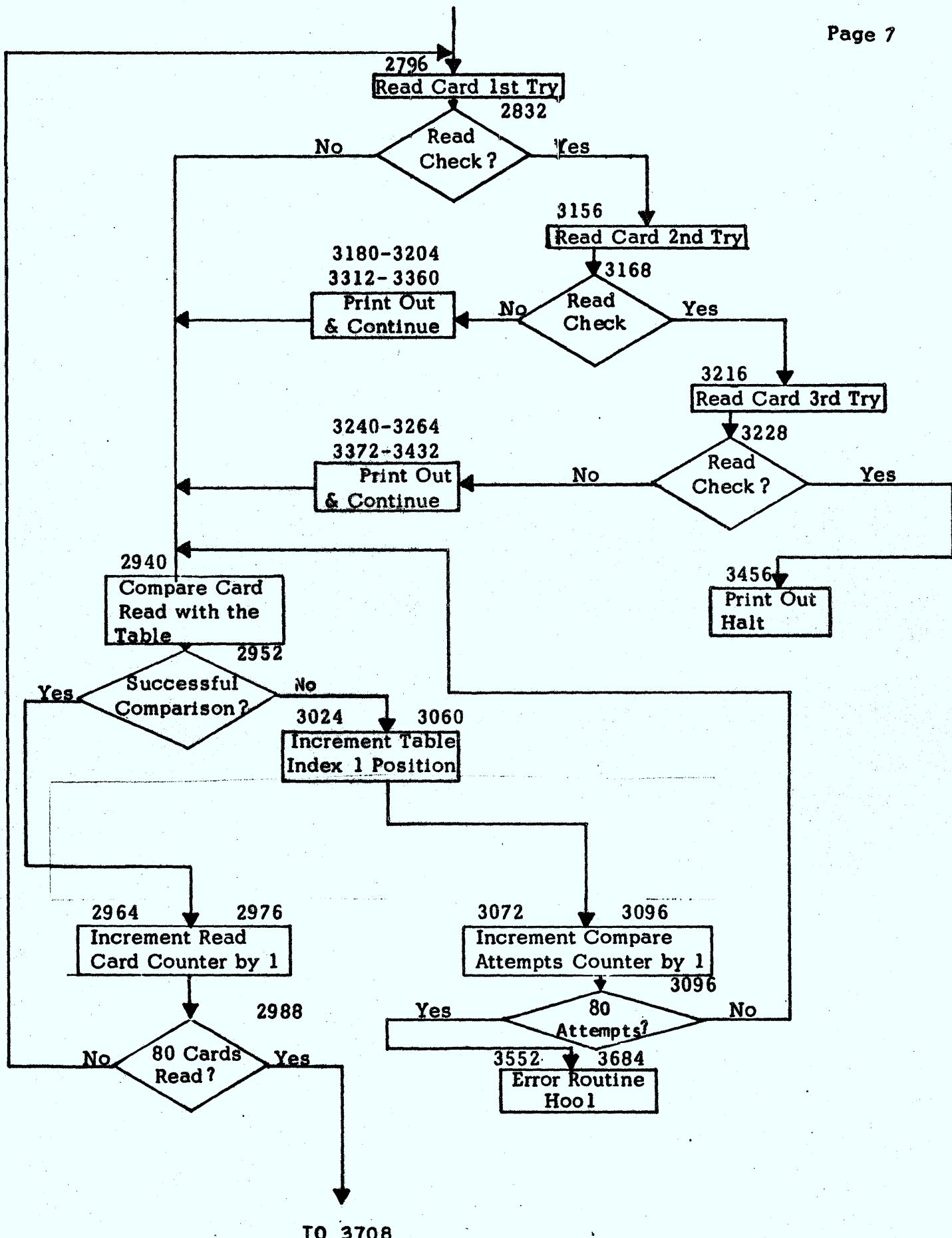
IF NO ETOS 1002 SUCCESSFUL.

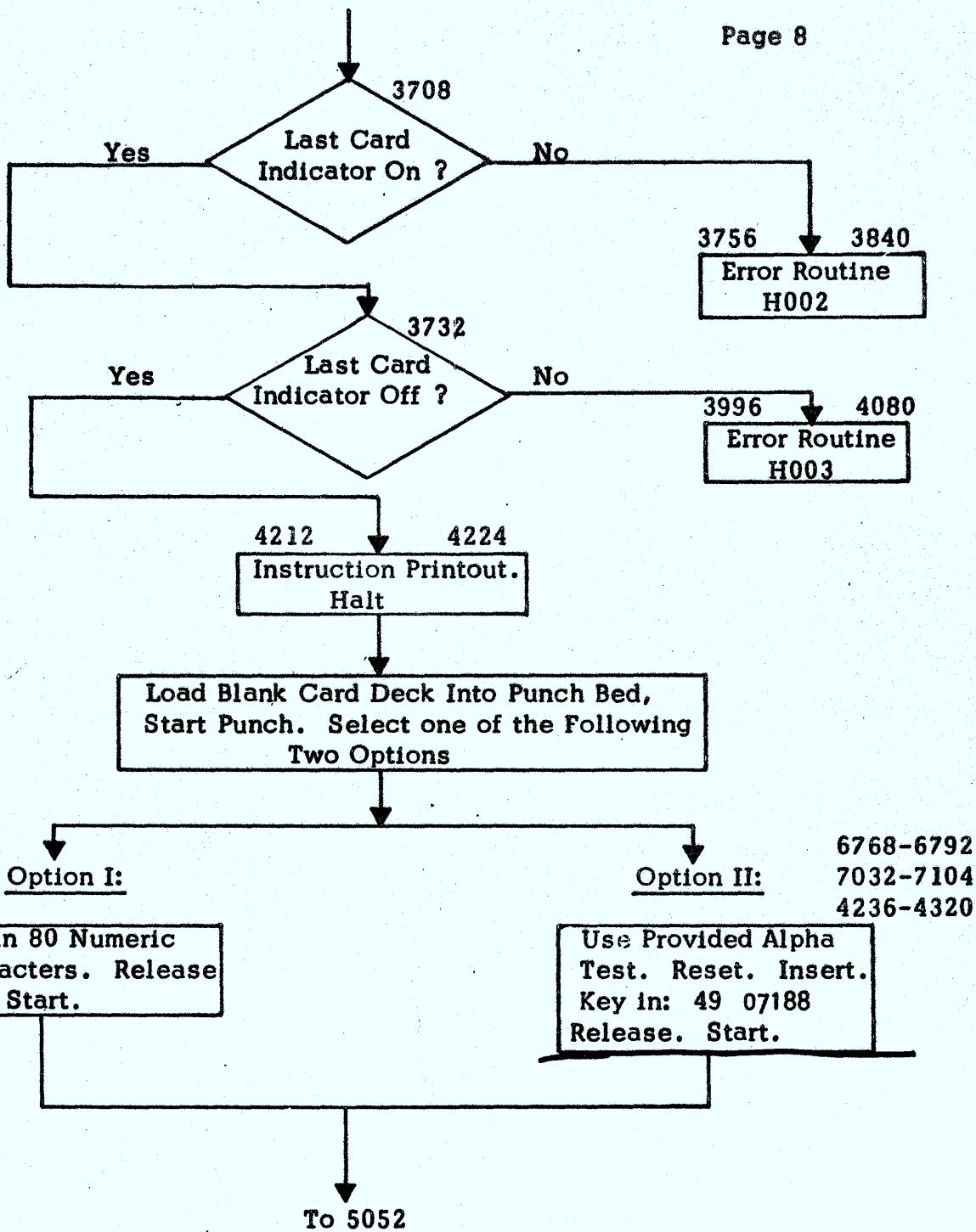
I002

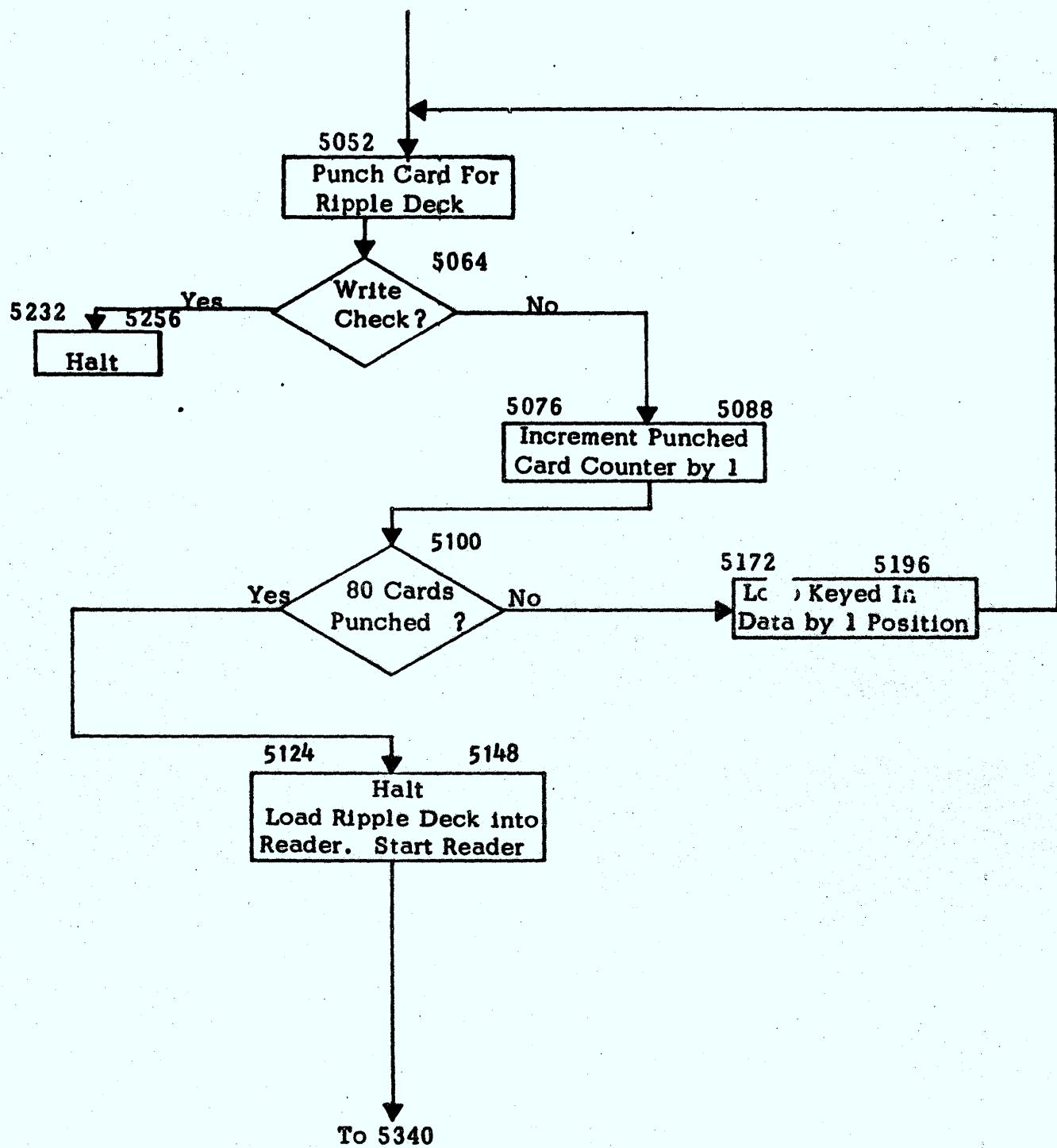


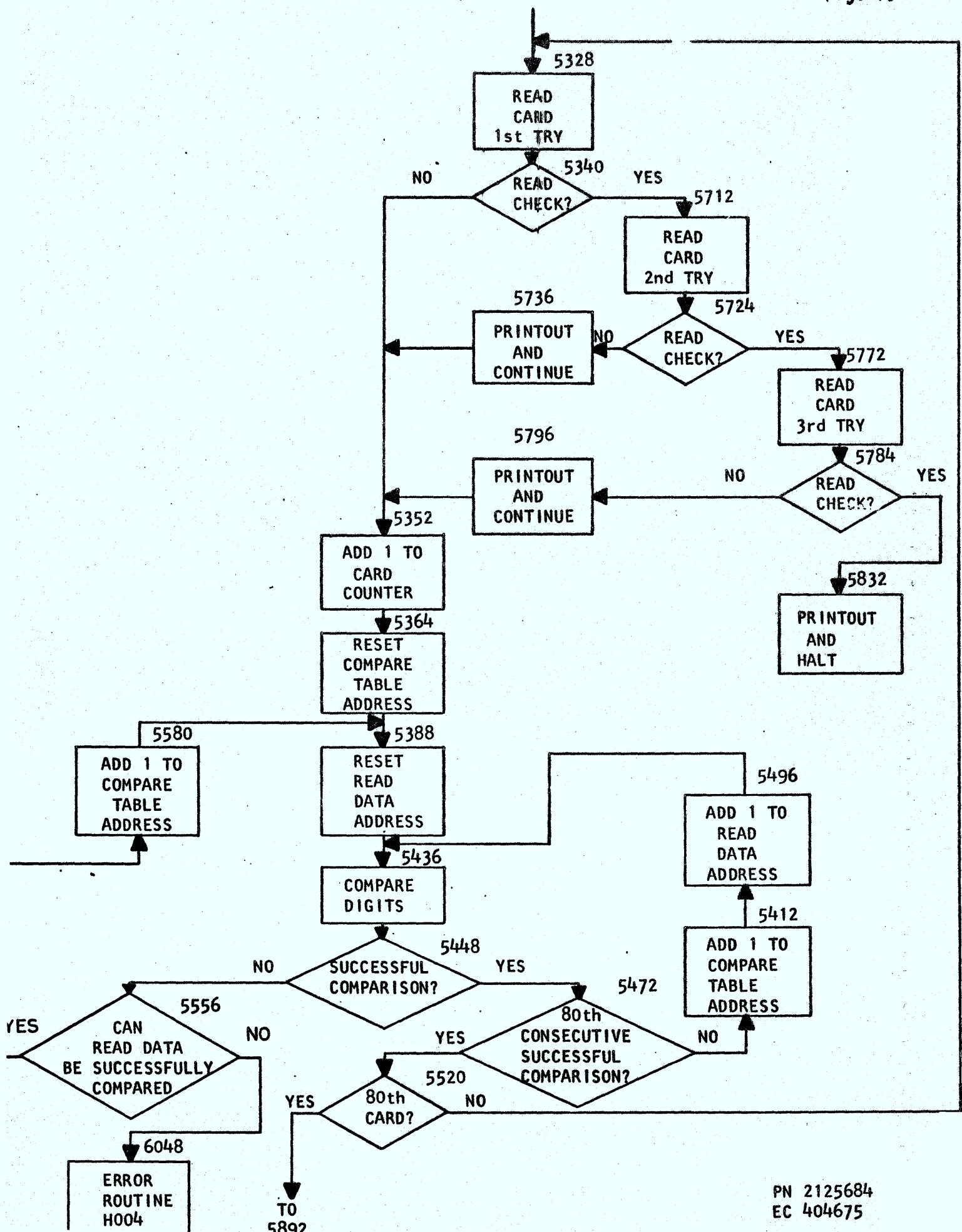
PN 2125684
EC 404839

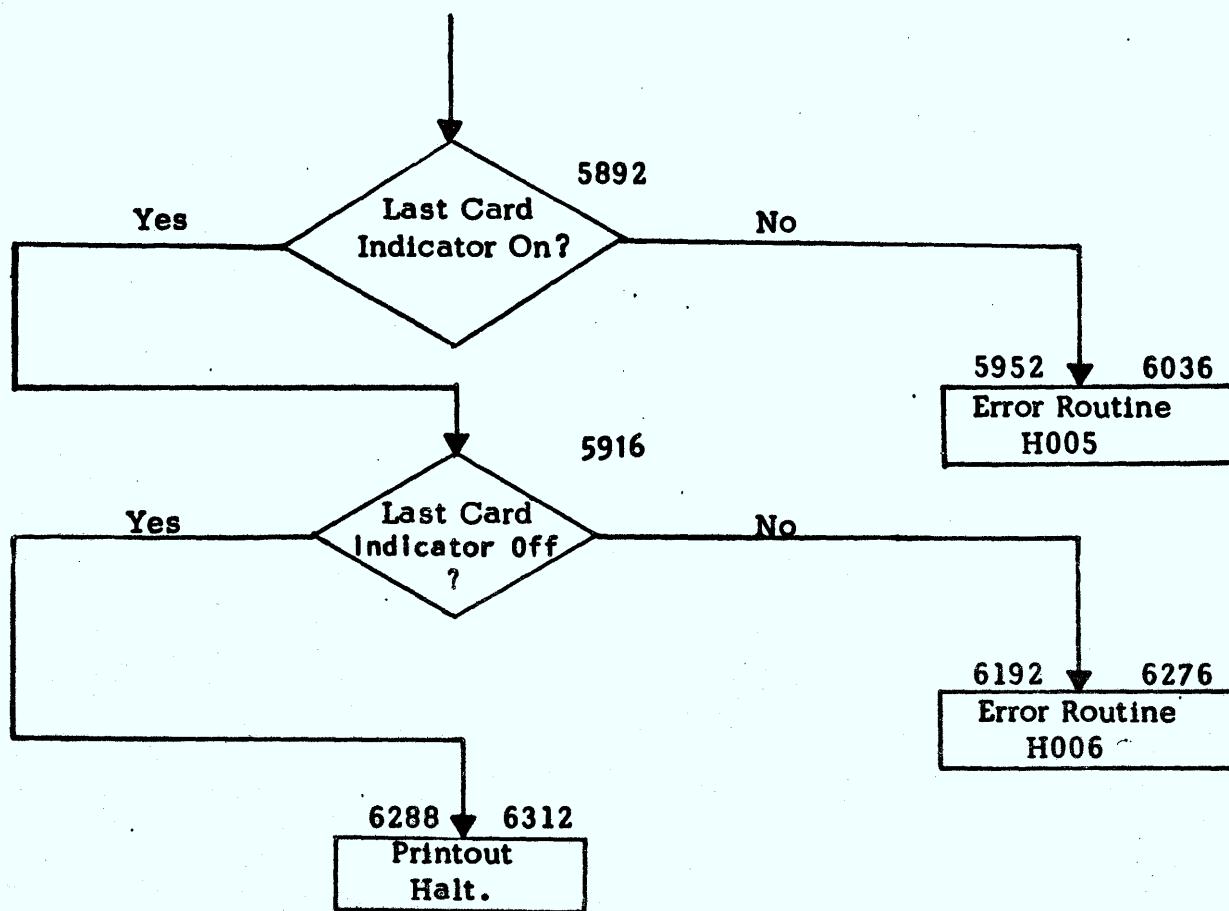












1 0 0 2

36	00060	00500	RN	1st Card
36	00100	00500	RN	
36	00160	00500	RN	
36	00220	00500	RN	
36	00280	00500	RN	
36	00340	00500	RN	2nd Card
36	00000	00500	RN	
49	00000		B	
00	00000	00000		3rd Card
10	20304	00020		
40	60800	03060		
90	21004	08021		
61	00500	15102		
00	60218	14200		4th Card
70	41128	20080		
61	42230	09081		
72	63000	00000		
00	50607	08090		
01	21416	18151		5th Card
81	12427	20242		
82	23635	20353		
04	54036	32484		
45	53249	46536		
04	84654	62754		6th Card
45	36271	80123		
45	67891	23456		
78	90234	56789		
01	34567	89012		
45	67890	12356		7th Card
78	90123	46789		
01	23457	89012		
34	56890	12345		
67	90123	45678		
36	00612	00500	RN	8th Card
11	00006	00060	AM	
14	00006	07212	CM	
46	00828	01200	BI	
49	00000	00000	B	

PN 2125684
EC 404675

1 0 0 2

0432

444

456

468

480

0492

504

516

528

540

0552

564

576

588

600

0612	62	66	7	1	56
624	55	0	≠6	266	
636	71	564		646	
648	0	≠	6266	72	
660	56	55	0	≠6266	

S	W	I	0
N	S	W	
I	O	F	F
S	W	2	
O	N	S	W

0672		72	5	64646	
684		0	≠626	6	73
696		5655		0	≠62
708	66	73		5646	
720	46	0	≠6	266	

2	O	F	F
S	W	3	
O	N	S	
W	3	O	F
F	S	W	

0732	74	565		5	0	≠
744	62	66	7	4	56	
756	46	46	0	≠	6245	
768	63	626		662		
780	46	5659		4970		

4	O	N		
S	W	4	0	
F	F	S	E	
T	S	W	S	
F	O	R	I	0

0792	56	72	6	34845	
0804	55	626		34159	
16	63	03	0	≠	
28	46	00852		00100	B I
40	47	00876		00100	BNI

0	2			
T	H	E	N	S
T	A	R	T	. ≠
Check S W 1 on				
Check S W 1 off				

0852	39	00613		00100	W A
64	49	00888			B
76	39	00631		00100	W A
88	46	00912		00200	B I
0900	47	00936		00200	BNI

S W 1 on				
S W 1 off				
Check S W 2 on				
Check S W 2 off				

0912	39	00651	00100	W A	S W 2 on	
	24	49	00948	B		
	36	39	00669	W A	S W 2 off	
	48	46	00972	B I	Check S W 3 on	
	60	47	00996	BNI	Check S W 3 off	
0972	39	00689	00100	W A	S W 3 on	
	84	49	01008	B		
	96	39	00707	W A	S W 3 off	
1008	46	01032	00400	B I	Check S W 4 on	
	20	47	01056	BNI	Check S W 4 off	
1032	39	00727	00100	W A	S W 4 on	
	44	49	01068	B		
	56	39	00745	W A	S W 4 off	
	68	39	00765	W A		
	80	34	00102	K	Carriage return	
1092	48			H		
1104	39	01213	00100	W A	Key in 80 alpha etc.	
	16	34	00102	K	carriage return	
	28	37	01933	R A	place 80 char in buffers	
	40	15	02093	TDM	set R M	
1152	34		00102	K		
	64	32	01932	S F		
	76	49	01332	B		
	88					
1200						
1212	52	4568	4955		Key in	
	24	7870	4153		80 Al	
	36	57	4841		Pha Ch	
	48	41	59414		aracte	
	60	59	62 6		rs To	
1272	47	45554	55941		Genera	
	84	63	45 5		te Rip	
	96	57	4345		ple De	
1308	43	5203	07		ck. ≠	
	20					
1332	31	02112	01932	T R	Generate work area	
	44	31	02472	01932	T R	Generate table'
	56	31	02632	01932	T R	look - up
	68	33	02632		C F	area
	80	46	01392	01400	B I	Turn off overflow

1392	46	01404	00600	B I	turn off R/C ind
1404	46	01416	00700	B I	turn off W/C ind
16	46	01428	00900	B I	turn off L/C ind
28	46	06612	01900	B I	check Any latch.
40	39	02113	00400	W A	punch ripple deck
1452	46	01620	00700	B I	W/C error
64	11	06347	01	A M	Add 1 to punched card counter
76	14	06347	80	C M	
88	47	01560	01200	BNI	ripple deck complete?
1500	16	06347	00	TFM	reset punched card counter
1512	39	01669	00100	W A	ripple deck has been punched etc.
24	34		00102	K	
36	48			H	
48	49	02796		B	
60	31	02110	02112	T R	
1572	26	02271	02111	T F	
84	33	02270		C F	
96	32	02112		S F	
1608	49	01440		B	
20	39	01765	00100	W A	W/C error etc.
1632	34		00102	K	
44	48			H	
56	49	01104		B	
68	59	49575	75345		Ripple
80		44454	352		Deck
1692	57	64554	34845		Punche
1704	44	535	64144		d Load
16		49556	356		Into
28	59	45414	44559		Reader
40		20 6	26341		- Sta
1752	59	6303	0≠		R T . ≠
64	66	2143	4559		W/C Er
76	59	5659	20		ror -
88	59	45626	34159		restar
1800	63	03 0	≠		t. ≠
1812					
24					
36					
48					
60					

1872
84
96
1908
20

1932
44
56
68
80

1992
2004
16
28
40

2052
64
76
88

0#

2112
24
36
48
60

Data xferred from 1932

2172
84
96
2208
20

2232
44
56
68
80

0#

2292
2304
16
28
40

Read in area for comp

2352
64
76
88
2400

**PN 2125684
EC 404618**

2412
24
36
48
60

2532
44
56
68
80

2592
2604
16
28
40

2652
64
76
88
2700

2712
24
36
48
60

2772				
84				0%
96	16	06371		00
2808	16	06407		00
20	37	02293		00500

**Reset read card ctr to 0
Reset attempts ctr to 0
R A read card 1st try**

2832	46	03156	006
44	32	02292	
56	26	02874	063
68	33		
80	32	02472	

R/C ind on ?
set flag in read in area
set up compare position
clear flag from last used table pos
set flag in comp pos

2892	26	06431	06419	T F	set up comp pos.
2904	26	06395	06383	T F	reset flag in table area
16	16	06407	00	TFM	reset comp attempts str to a
28	26	02946	06419	T F	
40	24	02631	02451	C	Compare read with table
2952	47	03024	01200	BNI	Compare?
64	11	06371	01	A M	add 1 to read completed cont
76	14	06371	80	C	80 completed?
88	47	02820	01200	BNI	read next card
3000	16	06371	00	TFM	reset read card counter to 0.
3012	49	03708		B	
24	26	03042	06395	T F	
36	33		-	C F	
48	11	06395	02	A M	Shift comp area in table
60	11	06431	02	A M	
3072	11	06407	01	A M	add to #comp attempted
84	14	06407	80	C	
96	46	03552	01200	B I	
3108	26	02946	06431	T F	
20	26	03138	06395	T F	
3132	32			S F	
44	49	02940		B	
56	37	02293	00500	R A	read card 2nd try
68	46	03216	00600	B I	
80	39	03313	00100	W A	
3192	34		00102	K	
3204	49	02844		B	
16	37	02293	00500	R A	read card 3rd try
28	46	03280	00600	B I	
40	39	03373	00100	W A	
3252	34		00102	K	
64	49	02844		B	
76					
80	39	03457	00100	W A	
3300	49	01524		B	
3312	59	2143	5655		R/C on
24		71626	3 63		1st t
36	59	6823	7255		ry, 2 n
48	44	635	968		d try
60	56	5203	0#		o k . #

PN 2125684
EC 404618

3372	59	2143	5655	R/C on
84		71626	3 10	1st +
96		72554	4 63	2nd T
3408	59	49456	223	ries,
20	73	5944	6359	3rd tr
3432	68	565	2030≠	Y ok. ≠
44				
56	59	2143	5655	R/C on
68	73	5944	63	3 rd t
80	59	68 2	0 59	ry - r
3492	45	53564	144	e load
3504	59	49575	75345	ripple
16		44454	352	deck
28	20	594	56263	- rest
40	41	59630	3 0≠	art . .≠
3552	46	03672	00100	B I
64	39	03685	00100	W A
76	34		00102	K
88	39	02293	00100	W A
3600	34		00102	K
				type out card contents
3612	39	02473	00100	W A
24	34		00102	K
36	48			H
48	46	02940	00200	B I
60	49	02964		B
				type out table contents
3672	47	03648	00300	BNI
84	48	70707	1 0≠	H
96				
3708	46	03732	00900	B I
20	49	03756		B
				Check L/C indicator
3732	46	03996	00900	B I
44	49	04212		B
56	46	03780	00100	B I
68	39	03793	00100	W A
80	47	03804	00300	BNI
				Check L/C ind off
3792	48	70707	2 0≠	H
3804	47	04164	00200	BNI
16	39	03925	00100	W A
28	34		00102	K
40	49	03708		B
				continue to numeric

3852

64

76

88

3900

3912

24	53	2143	4955
36	44	49434	16356
48	59	624	85664
60	53	44 4	245

L/C in
dicato
r shou
ld be

3972

56

84

96	46	04020	00100
4008	39	04033	00100
20	47	04044	00300

on . ≠

H003 L/C not off

4032

48

44

56

68

80

49

70707	3	0≠
04164	00200	BNI
04093	00100	W A
	00102	K
03708		B

H

BNI

W

A

K

B

4092

53

44

16

28

40

49

2143	4955
49434	16356
555	663
64595	54955
564	64603

L/C in
dicato
r not
turnin
g off.

4152

0≠

64

76

88

4200

39

24

36

48

60

04357	00100
04224	
	W A
	B

≠

4212

41

84

96

4308

31

20

04333	00100
	00102
04512	00100
04592	0000≠
	TDM
04772	00102
04956	

read in numeric
set R M

NOP

T R

T R

T R

B

4332	41	53574	841	Alpha
44	56	52 2	0	o k -
5t	2	4568	4955	key in
68		7870	5564	80 nu
80	54	45594	943	meric
4392	43	48415	94143	charac
4404	63	45596	2 63	ters t
16	56	474	55545	o gene
28	59	41634	5 59	rate r
40	49	57575	345	ipple
4452	44	45435	203	deck.
64	0#			#
76				
88				
4500				
4512				
24				keyed in
36				numeric
48				data
60				
4572				
84		#		
96				
4608				
20				
4632				xferred data
44				
56				
68		#		
80				
4692				
4704				
16				
28				
40				table
4752				area
64				
76				
88				
4800				

4812
24
36
48
60
≠

4872
84
96
4908
20

read in area

4932
44
56 46 04968 00600 B I
68 46 04980 00700 B I
80 46 04992 00900 B I

4992 46 05004 01400 B I
5004 46 05016 01600 B I
16 46 05028 01700 B I
28 46 06588 01900 B I
40 46 06708 00900 B I

5052 38 04596 00400 W N punch ripple deck
64 46 05232 00700 B I W/C error
76 11 06443 01 A M
88 14 06443 80 C M
5100 47 05172 01200 BNI

5112 16 06443 00 TFM
24 39 01669 00100 W A ripple deck punched etc
36 34 00102 K
48 48 H
60 49 05304 B

5172 31 04595 04596 T R loop numeric work
84 25 04675 04595 T D area for ripple deck
96 49 05052 B

5208
20

PN 2125684

EC 404675

5232	39	01765	00100	W A	W/C error etc
44	34		00102	K	
56	48			H	
68	39	04357	00100	W A	
80	49	04248		B	
05292					
05304	16	05528	00	T F M	Reset Card Counter
05316	16	05375	04691	T F M	Set Compare Table Address
05328	36	04872	00500	R N	Read a Card
05340	46	05712	00600	B I	Check for Read Check
05352	11	05528	01	A M	Add 1 to Card Counter
05364	16	05435	04691	T F M	Set Compare-to Table Address
05376	16	05480	00	T F M	Reset
05388	16	05411	04872	T F M	Set Read Table Address
05400	25	05444		T D	Transmit Read Digit
05412	11	05435	01	A M	Add 1 to Compare-to Table Address
05424	25	05447		T D	Transmit Compare-to Digit
05436	14	05444	00 00	C M	Compare Digits
05448	47	05556	01200	B N I	
05460	11	05480	01	A M	Add 1 to successful Compare Counter
05472	14	05480	00 80	C M	80th Consecutive Successful Compare?
05484	46	05520	01200	B I	
05496	11	05411	01	A M	Add 1 to Read Digit Address
05508	49	05400		B	
05520	14	05528	00 80	C M	80th Card?
05532	47	05316	01200	B N I	
05544	49	05892		B	
05556	14	05435	04772	C M	Can Card be Compared?
05568	46	06048	01100	B I	
05580	11	05375	01	A M	Add 1 to Compare-to Table Address

5592	49	05364	B		
5604					
16					
28					
40					
 5652					
64					
76					
88					
5700					
5712	36	04872	00500	R N	read card 2st try
24	46	05772	00600	B I	
36	39	03313	00100	W A	
48	34		00102	K	
60	49	05352		B	
 5772	36	04872	00500	R N	read card 3rd tr.
84	46	05832	00600	B I	
96	39	03373	00100	W A	
5808	34		00102	K	
20	49	05352		B	
 5832	39	03457	00100	W A	R/C on 3rd try reload etc.
44	49	05136		B	
56					
68					
80					
 5892	46	05916	00900	B I	Check L/C ind
5904	49	05952		B	
16	46	06192	00900	B I	Check L/C ind off
28	49	06288		B	
40					
 5952	46	05976	00100	B I	
64	39	05989	00100	W A	
76	47	06000	00300	BNI	
88	48	70707	5 0	H	
6000	47	06288	00200	BNI	
 6012	39	03925	00100	W A	
24	34		00102	K	
36	49	05892		B	
48	46	06168	00100	B I	
60	39	06181	00100	W A	type out H004

6072	34		00102	K	
84	38	04872	00100	W N	type out card contents
96	34		00102	K	
6108	38	04692	00100	W N	Type out table contents
20	34		00102	K	
6132	48			H	
44	46	05436	00200	B I	
56	49	05520		B	
68	47	06144	00300	B I	
80	48	70707	4 0≠	H	
6192	46	06116	00100	B I	
6204	39	06229	00100	W A	
16	47	06240	00300	BNI	
28	48	70707	6 0≠	H	
40	47	06288	00200	B N I	
6252	39	04093	00100	W A	
64	34		00102	K	
76	49	05892		B	
88	39	06529	00100	W A	
6300	34		00102	K	
6312	48			H	
24					
36			00		alpha punched card counter
48					
60			00		alpha read card counter
6372			02472		1st mem pos of alpha table look up
84			02472		
96			00		alpha comp attempted
6408			02631		1st comp area alpha
20					
6432			00		card count for num punch cards
44			04692		1st mem pos num of table look up
56			04692		
68			04771		1st num comp area
6492			00		counter num for comp attempts
6504			00		card read num counter
16					
28	49	46 5	556		if no
40	45	63566	2 49		etos I
6552	56	7072	6264		0 0 2 su
64	43	43456	26246		c c e s s f
76	64	5303	0≠		u l. ≠
6588	39	06723	00100		
6600	47	06624	00600		

6612	39	06753	00100	W A
24	47	06648	00700	B N I
36	39	06771	00100	W A
48	47	06672	01600	B N I
60	39	06789	00100	W A
6672	47	06696	01700	B N I
84	39	06813	00100	W A
96	48			H
6708	39	06837	00100	W A
20	48	41556	80044	H
				Any D
6732	41	63410	04348	ATA CH
44	52	03000	#5944	K. # RD
56	00	43485	20300	CHK.
68	0#	66590	04348	# WR CH
80	52	03000	#5442	K. # MB
6792	59	20450	04348	R-E CH
6804	52	03000	#5442	K. # MB
16	59	20560	04348	R-O CH
28	52	03000	#5321	K. # L/
40	43	00565	5030#	C ON.#

6852	41	42434	44546	
64	47	48495	15253	
76	54	55565	75859	
88	62	63646	56667	Dummy alpha
6900	68	69 7	07172	read in data

6912	73	74757	67778	
24	79	03041	01314	
36	20	21232	43334	
48	41	42434	44546	
60	47	48495	15253	

6972	54	55565	75859	
84	62	63646	56667	
96	68	69 5	07172	
7008	73	740#		
	20			

07032	01	23456	78901	
44	23	45678	90123	
56	45	67890	12345	
68	67	89012	34567	
80	89	01234	56789	

dummy numeric
read in data

07092	01	23456	78901	
7104	23	45678	9#	

16
28
40

07152	31	01932	06852	T R
64	49	01140		B
76				
88	31	04512	07032	T R
07200	49	04248		B

IO03– Card I/O Reliability Diagnostic


```
//=====
//  
// IO03 - Card I/O Reliability Diagnostic  
//  
// Program Switch settings:  
//  
// PS1: ON - Punch only  
// OFF - Read and punch  
// PS2: ON - Read only  
// OFF - Read and punch  
// PS3: ON - Don't print errors  
// OFF - Print errors  
// PS4: ON - Delay constant  
// OFF - Delay changes  
//  
// Check switches settings:  
//  
// DISK I/O - STOP  
// PARITY - STOP  
// I/O - PROGRAM  
// O'FLOW - PROGRAM  
//  
// Start addresses:  
//  
// 00652 - Full test  
//  
// Directions:  
//  
// 1. Load IO03 diagnostic  
// 2. Turn on PROGRAM SWITCH 1  
// 3. Press START  
// 4. Press START  
// 5. Verify that PUNCH NO FEED light is lit  
// 6. Insert empty card deck into punch  
// 7. After ~ 1 minute press STOP  
// 8. Remove cards from punch  
// 9. Turn off PROGRAM SWITCH 1  
// 10. Press RESET  
// 11. Press INSERT  
// 12. Type 4900652  
// 13. Press RELEASE  
// 14. Press START  
// 15. Verify that READER NO FEED light is lit  
// 16. Insert newly punched cards in reader  
// 17. Verify that PUNCH NO FEED light is lit  
// 18. Insert empty card deck into punch  
//=====
```


Sample Output - IO03

1622 CARD READER-PUNCH DELAY IO03.

SW 1 ON= PUNCH ONLY. SW 2 ON= READ AND COMPARE ONLY.

SW 3 ON= BY-PASS ETOS. SW 4 ON= STOP CHO-CHO DELAY CHANGE.

4900652

READER CARD COUNTER = ~~00000200~~

PUNCH CARD COUNTER = ~~00000298~~

NO. 2125682
SHEET 0
OF 20

DIAGNOSTIC TEST

TITLE 1622 CARD INPUT/OUTPUT DIAGNOSTIC TEST (INTERLEAVING) - 1003
MACH. TYPE 1620 BY JHM APPR. G.I.A. DATE 4-11-62

ENGINEERING CHANGE HISTORY

E/C NO.	DATE	SHEETS AFFECTED
404674	11-4-61	1-20
404675	4-11-62	1,2,3,4,5,6,11,12,13, 16,17,18

E/C NO.	404674	404675					
DATE	11-4-61	4-11-62					

A. SCOPE

This is a Reliability Test Program for the 1620 - 1622 System. Cards are punched with a ripple pattern using alphanumeric characters which are read back in and compared to stored data.

Basically, the program operates in the following sequence:

- a) Check for Any Data Checks.
- b) Read a card.
- c) Check 1st character.
- d) Compare card information.
- e) Same as (a) through (d) above.
- f) Punch card.
- g) Loops back to (a).

There is a delay routine (cho-cho) designed to create every possible timing condition between reading and punching. The delay starts at approximately ten seconds between cards and progresses to zero delay (maximum reading and punching speed). One complete cycle of the delay (maximum delay to zero and back to maximum) takes approximately 30 minutes.

B. OPERATING INSTRUCTIONS

1. Reset 1620.
2. Place Program Deck in Card Reader Hopper.
3. Push "Load" P. B. on 1622.
4. 1620 will HALT after typing out Heading.
5.
 - a. Set Program SWS for desired options.
 - b. Ready the punch.
(Note: To punch the initial ripple deck, place SW 1 ON and Start. After punching about 200 cards, press the SIE Key. Insert 4900652, Release, place SW 1 OFF and proceed.)
 - c. Place punched ripple cards in card reader hopper. Ready the reader.
6. Push START on 1620 to begin Program.
7. Place punched cards in reader hopper to be read and checked.

Switch Settings (suggested)

Data Check	= Program
I/O Check	= Program
Arith Check	= Must be Program

PN 2125632
EC 404675

SW 1 = OFF (After punching ripple deck)
SW 2 = OFF
SW 3 = OFF
SW 4 = OFF

SW 1 OFF Read and punch
ON Punch only

SW 2 OFF Read and Punch
ON Read only

SW 3 OFF Errors typed out
ON Bypass E'TO (for trouble shooting)

SW 4 OFF Delay changes
ON Delay remains constant

Error Type-Outs (ETOs):

There are several Data Checks made throughout the program. When an error has been detected, a type-out (if program SW 3 is OFF) will give the following information about the error:

- a) The memory location of the instruction that detected the error.
- b) Where possible the correct information will be typed out followed by the information containing the error.

0724 READ CHECK
00736 WRITE CHECK
00748 O/F INDICATOR ON
00760 MBR - E CHECK
00772 MBR - O CHECK
00784 MAR CHECK

Error Type-Outs (ETOs) (cont'd)

The above type-outs are due to a data check condition at the start of the read routine. All of these indicators should be OFF at the start of each Program pass.

00976 READ CHECK 1ST READ
01012 " " 2ND "
01048 " " 3RD "

These ET0s are due to a read check when the 1622 transfers the card information to the 1620. Three attempts are made to transfer each card if necessary. Check 1620 I/O translation circuit and check buffer (1622) for correct data.

01096 MBR - E CHECK AFTER CARD READ
01108 MBR - O CHECK AFTER CARD READ

These ET0s are due to an MBR check and indicate trouble in the I/O translator or memory circuit (1620). The checks are made directly after a card is read by the 1620.

01120 READ-IN AREA 0/F

The 1620 received more than 80 alpha characters from the 1622.

01204 NO 1ST COMPARE

The first character in the card Read-In was not one of the 48 alpha characters used by this program. Check to see if card contains correct data.

01360 MBR - E CHECK AFTER 1ST COMPARE
01372 MBR - O CHECK AFTER 1ST COMPARE

01564 MBR - E CK BEFORE COMPARE.
01576 MBR - O CK BEFORE COMPARE.

Due to data check while performing 1st compare routine.
Check 1620 for correct internal transfer operation.

01624 E/Z TGR OFF COMPARE DATA FOLLOWS.

The correct card data will be typed from STORED DATA followed by the data read-in from the card.

This ET0 is due to the information read-in not corresponding to the stored data. Check card and buffer (1622) for correct data.

01636 MBR - E CHECK AFTER COMPARE, DATA FOLLOWS.
01648 MBR - O CHECK AFTER COMPARE,DATA FOLLOWS.

Data typed-out is same as for E/Z TGR OFF (above). Caused during card compare routine by internal transfer operation (1620)

OPERATING INSTRUCTIONS (cont'd)

Page -4-

Error Type-Outs (ETOs) (cont'd)

01960 WRITE CHECK AFTER PUNCH.

Checked after each punch routine, Check punched card for correct data.

The complete normal typeout information will be as follows:

1622 CARD READER-PUNCH DELAY 1003.
SW 1 ON= PUNCH ONLY. SW 2 ON= READ AND COMPARE ONLY.
SW 3 ON= BY-PASS ETOs. SW 4 ON= STOP CHO-CHO DELAY CHANGE.

MEMORY ADDRESS ALLOCATIONS

0 0 0 0 0	→	0 0 0 9 6	Load Card and Branch Instructions
1 0 0	→	3 9 9	Math. Tables
4 1 2	→	5 9 1	Load Program
6 5 2	→	2 3 8 0	Main Program
2 5 1 2	→	3 8 0 7	ETO Routines
3 8 0 9	→	5 0 6 5	ETO Data
5 1 0 1	→	5 2 5 9	Card Counter T. O. Data
5 2 6 1	→	5 5 9 9	Heading TO Data
5 6 0 5	→	5 9 2 5	Card Compare Data
5 9 3 5	→	6 1 6 5	Read-In Clear Data
6 1 7 5	→	6 4 0 5	1st Character Compare Data
6 4 1 5	→	6 7 3 5	Compare Working Area
6 7 4 5	→	6 9 7 5	Read-In Area

1003 LOAD PROGRAM

00000 to 00059

36	00060	00500
36	00100	00500
36	00160	00500
36	00220	00500
36	00280	00500

Card
#1

00060 to 00095

36	00340	00500
36	00000	00500
49	00000	00000

Card
#2

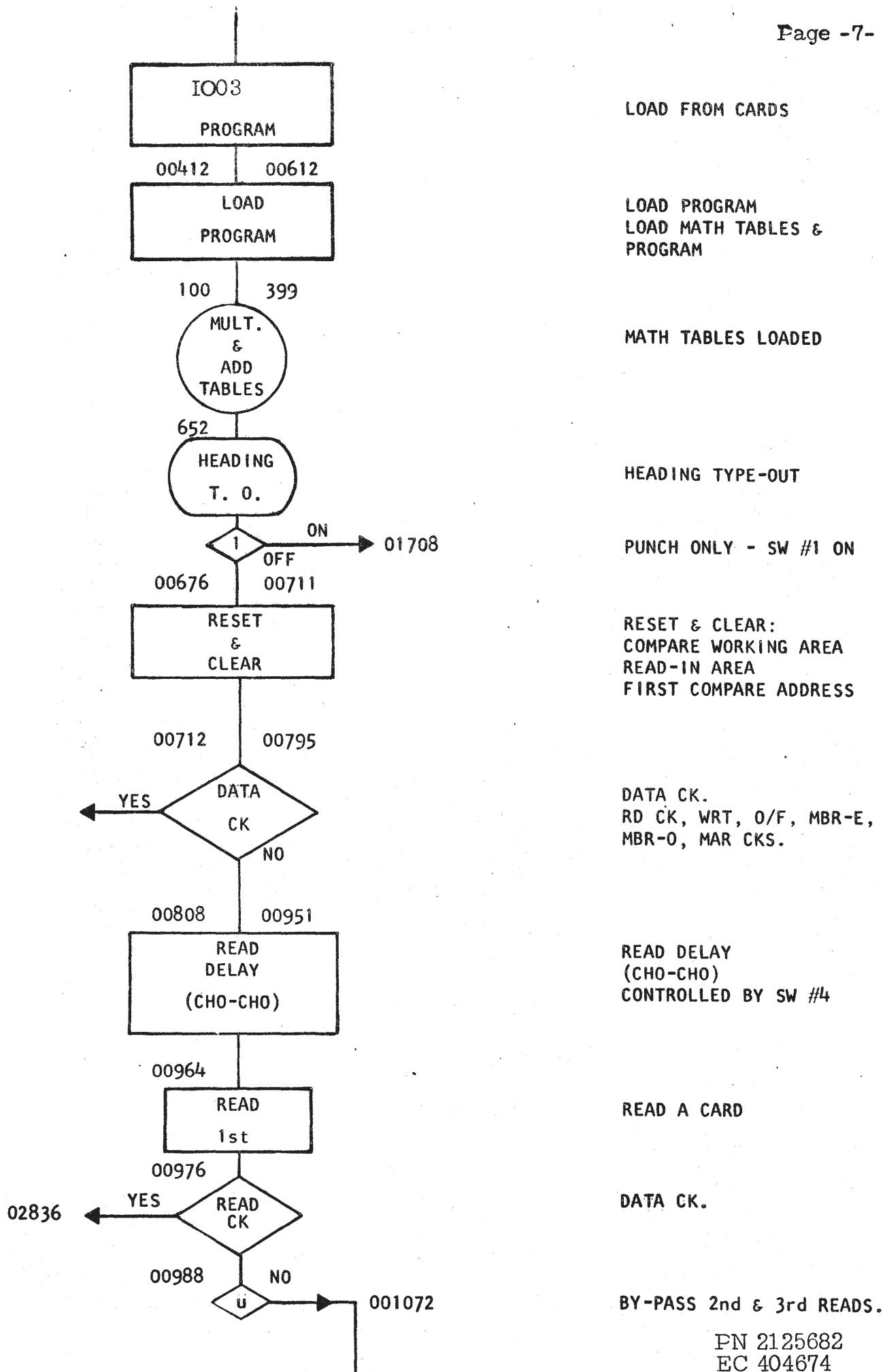
00000 to 00059

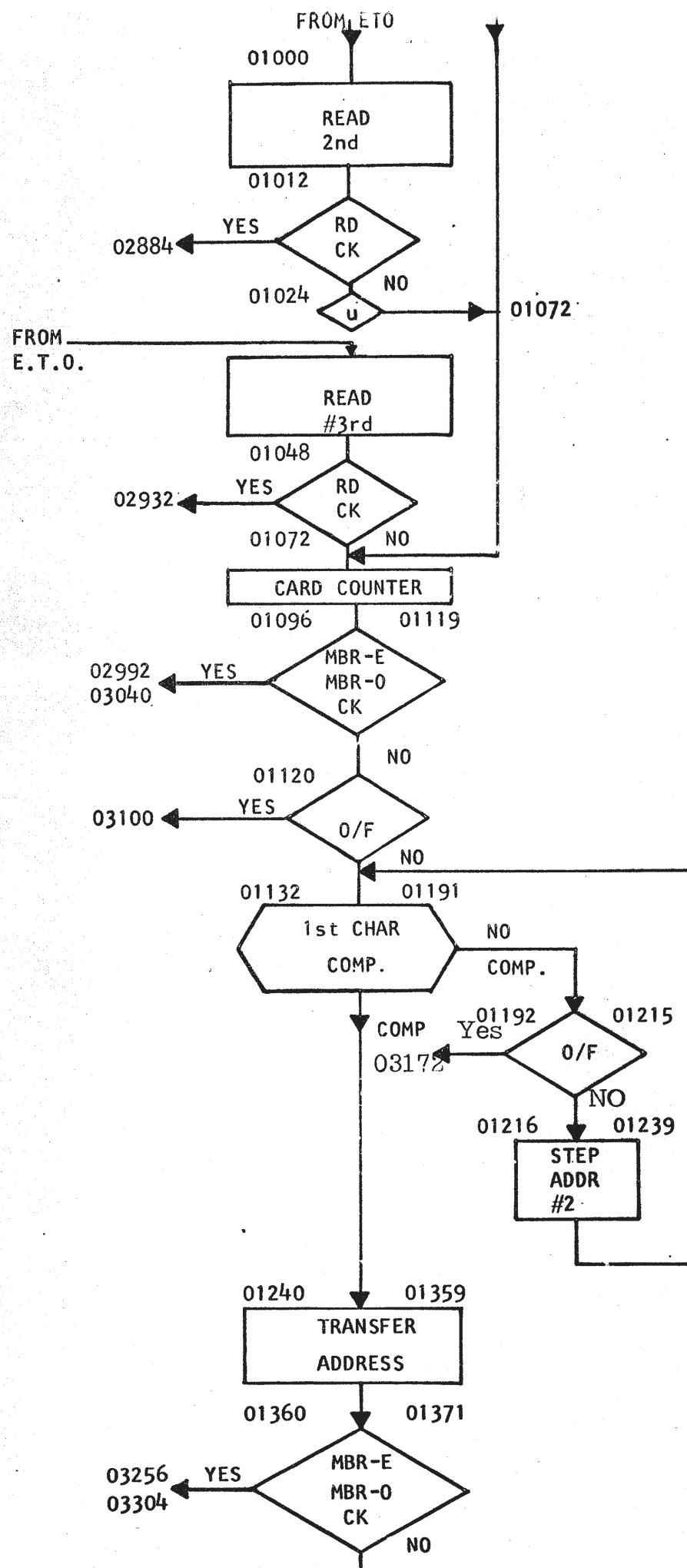
36	00652	00500
11	00006	00060
14	00006	06772
46	00652	01200
49	00000	

Card
#8

First and second Load Cards load the math tables and the Program Load Card. (Cards 3 through 7 contain the math tables.)

Eighth Load Card contains instructions for loading core storage.



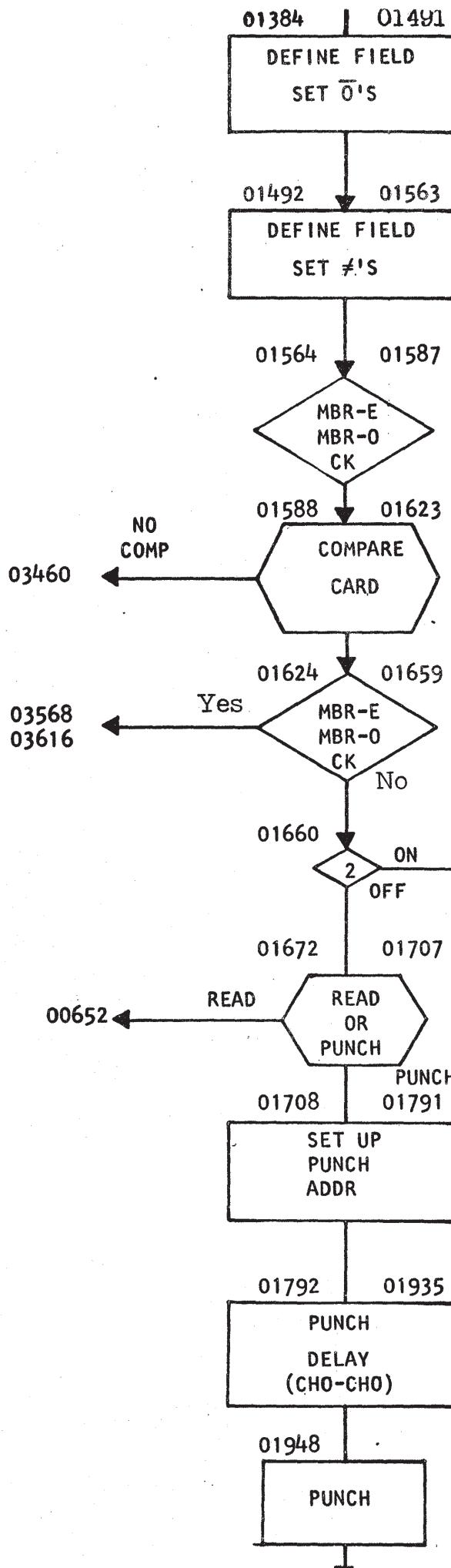


BY-PASS 3rd READ

STEP READER CARD COUNTER

DATA CK.

O/F IF MORE THAN 80
CHARACTERS READ1st CHARACTER IS CHECK
WITH DATA IN MEMORY TO
SET UP AN ADDRESS TO
COMPARE THE CARD READ
IN FOR CORRECT DATA



SET 0's IN COMPARE WORKING AREA TO DEFINE FIELD FOR "CARD COMPARE"

SET #'S IN COMPARE WORKING AREA TO DEFINE FIELD FOR "CARD COMPARE"

DATA CK.

COMPARE CARD READ WITH KNOWN DATA IN MEMORY

DATA CK

READ ONLY (SW #2 ON)

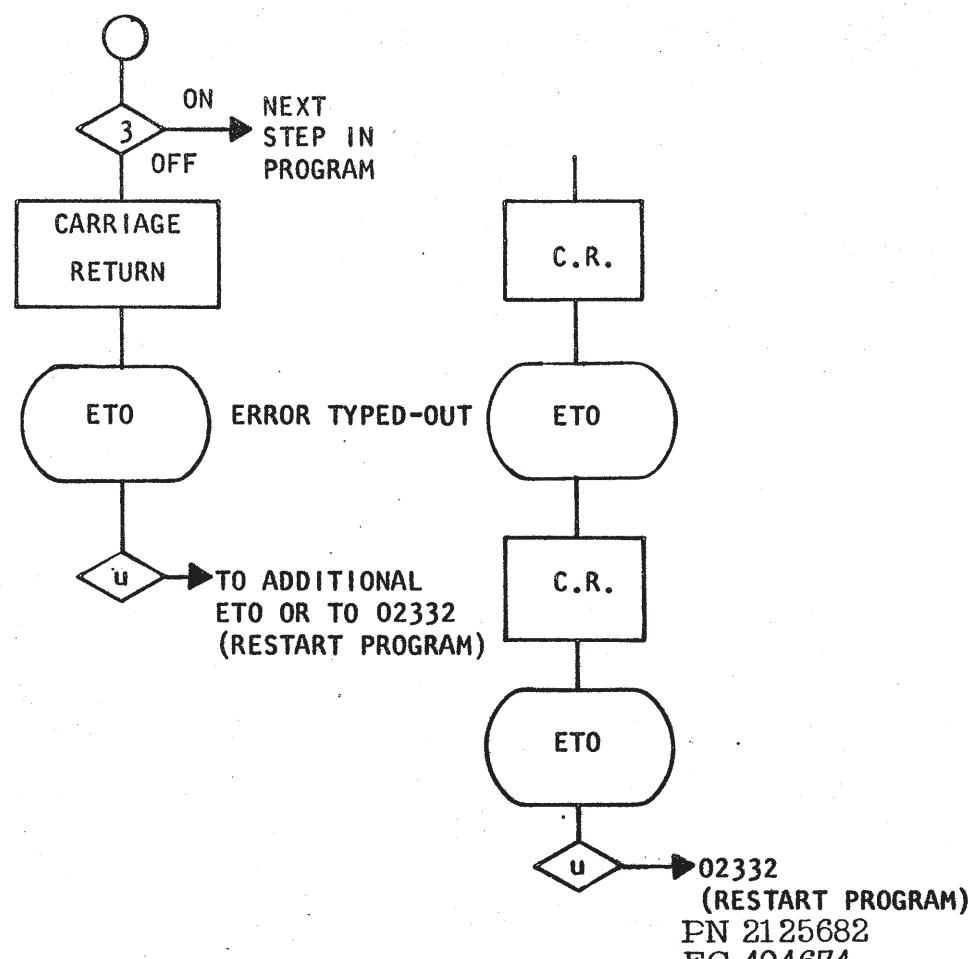
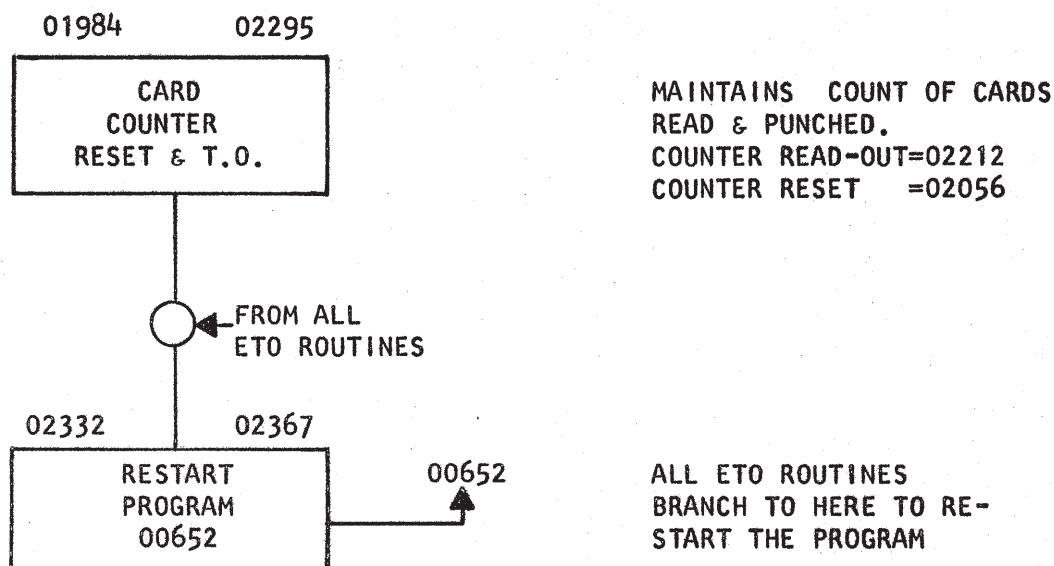
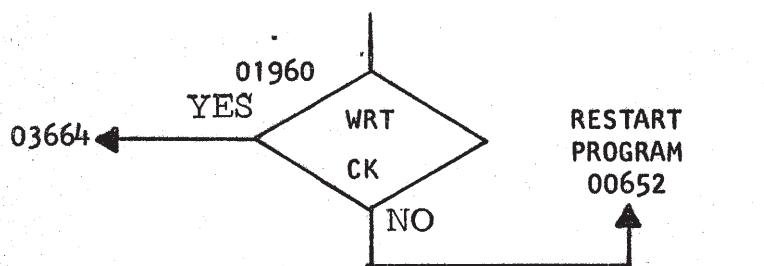
DETERMINE IF READ OR PUNCH OPERATION IS NEXT

STEP PUNCH ADDRESS TO PRODUCE RIPPLE DECK

PUNCH DELAY

(CONTROLLED BY SW #4)

PUNCH RIPPLE DECK FROM DATA STORED IN MEMORY
PN 2125682
EC 404674



PN 2125682
EC 101671

00652	49	03712	00000	NOP	
00664	46	01708	00100	BI	CHECK SW #1
00676	26	06735	05925	TF	RESET COMPARE AREA
00688	26	06975	06165	TF	CLEAR READ-IN AREA
00700	16	01167	06175	TFM	RESET FIRST COMP. ADDRESS
00712	41	00000	00000	NOP	
00724	46	02512	00600	BI	READ CHK
00736	46	02572	00700	BI	WR CHK
00748	46	02620	01400	BI	OVERFLOW
00760	46	02668	01600	BI	MBR-EVEN
00772	46	02728	01700	BI	MBR-ODD
00784	46	02776	00800	BI	MAR
00796	41	00000	00000	NOP	
00808	41	00952	00000	NOP	
00820	11	00949	00101	AM	ADD TO DELAY COUNTER
00832	46	00856	01400	BI	OVERFLOW?
00844	49	00820	00000	B	
00856	16	00949	00000	TFM	RESET DELAY COUNTER
00868	46	00952	00400	BI	CHECK SW #4
00880	11	00831	00002	AM	SHORTEN DELAY
00892	46	00916	01400	BI	OVERFLOW?
00904	49	00952	00000	B	
00916	16	00831	00101	TFM	RESTORE LONG DELAY
00928	49	00952	00000	B	
00940	41	00000	00000	NOP	
00952	41	00000	00000	NOP	
00964	37	06745	00500	RA	READ A CARD
00976	46	02836	00600	BI	READ CHK?
00988	49	01072	00000	B	BYPASS 2nd & 3rd tries
01000	37	06745	00500	RA	READ 2nd ATTEMPT
01012	46	02884	00600	BI	READ CHK?
01024	49	01072	00000	B	BYPASS 3rd try
01036	37	06745	00500	RA	READ 3rd ATTEMPT
01048	46	02932	00600	BI	READ CHK?
01060	49	01072	00000	B	
01072	11	02030	00001	AM	STEP CARD RD COUNTER
01084	43	02044	02023	BD	
01096	46	02992	01600	BI	MBR-EVEN CHK?
01108	46	03040	01700	BI	MBR-ODD CHK?
01120	45	03100	06909	BNR	
01132	41	00000	00000	NOP	
01144	32	06744	00000	SF	
01156	24	06745	06175	C	COMPARE 1st CHARACTER
01168	47	01192	01200	BNI	
01180	49	01240	00000	B	
01192	14	01167	06275	CM	
01204	46	03172	01200	BI	
01216	11	01167	00002	AM	STEP COMPARE ADDRESS
01228	49	01156	00000	B	
01240	41	00000	00000	NOP	
01252	25	01320	01167	TD	
01264	25	01319	01166	TD	TRANSFER ADDRESS
01276	25	01318	01165	TD	
01288	11	01320	00240	AM	STEP TO COMPARE DATA ADD.
01300	49	01336	00000	B	

01312	41	00060	00000	NOP	
01324	41	00060	00000	NOP	
01336	26	01332	01320	TF	
01348	26	03514	01320	TF	
01360	46	03256	01600	BI	MBR-EVEN CHK?
01372	46	03304	01700	BI	MBR-ODD CHK?
01384	41	00000	00000	NOP	
01396	12	01320	00001	SM	
01408	26	01426	01320	TF	
01420	32	00000	00000	SF	
01432	12	01320	00001	SM	
01444	26	01462	01320	TF	
01456	32	00000	00000	SF	
01468	41	00000	00000	NOP	
01480	11	01332	00159	AM	
01492	26	01510	01332	TF	
01504	15	00000	00007	TDM	
01516	11	01332	00001	AM	
01528	26	01546	01332	TF	
01540	15	00000	00007	TDM	
01552	12	01332	00002	SM	
01564	46	03352	01600	BI	
01576	46	03400	01700	BI	
01588	41	00000	00000	NOP	
01600	26	01618	01332	TF	
01612	24	06611	06903	C	COMPARE
01624	47	03460	01200	BNI	
01636	46	03563	01600	BI	
01648	46	03616	01700	BI	
01660	46	00652	00200	BI	CHECK SW #2
01672	11	01706	00050	AM	
01684	43	01703	01704	BD	READ-READ-PUNCH
01696	49	00652	00000	B	COUNTER
01708	41	00000	00000	NOP	
01720	11	01954	00002	AM	STEP PUNCH ADDRESS
01732	14	01954	05705	CM	
01744	46	01768	01200	BI	
01756	49	01780	00000	B	
01768	16	01954	05605	TFM	RESTORE PUNCH ADDRESS
01780	16	01707	00000	TFM	RESET RD-RD-PCH COUNTER
01792	41	01936	00000	NOP	
01804	11	01933	00101	AM	ADD TO DELAY COUNTER
01816	46	01840	01400	BI	OVERFLOW?
01828	49	91804	00000	B	
01840	16	01933	00000	TFM	RESET DELAY COUNTER
01852	46	01936	00400	BI	CHECK SW #4
01864	11	01815	00004	AM	SHORTEN DELAY
01876	46	01900	01400	BI	OVERFLOW?
01888	49	01936	00000	B	
01900	16	01815	00101	TFM	RESTORE LONG DELAY
01912	49	01936	00000	B	
01924	41	00000	00000	NOP	
01936	41	00000	00000	NOP	
01948	39	05605	00400	WA	PUNCH A CARD
01960	46	03664	00700	BI	WRITE CHK?
01972	11	02042	00001	AM	STEP CARD PCH COUNTER
01984	43	02044	02035	BD	

01996	41	00000	00000	NOP
02008	49	00652	00000	B
02020	41	00000	0000#	NOP
02032	41	00000	0000#	NOP
02044	41	00000	00000	NOP
02056	34	00000	00102	K
02068	39	05101	00100	WA
02080	38	02023	00100	WN
02092	34	00000	00102	K
02104	39	05151	00100	WA
02116	38	02035	00100	WN
02128	34	00000	00102	K
02140	39	05201	00100	WA
02152	41	00000	00000	NOP
02164	26	02030	02210	TF
02176	26	02042	02210	TF
02188	49	02212	00000	B
02200	41	00000	0000#	NOP
02212	41	00000	00000	NOP
02224	34	00000	00102	K
02236	39	05101	00100	WA
02248	38	02023	00100	WN
02260	34	00000	00102	K
02272	39	05151	00100	WA
02284	38	02035	00100	WN
02296	41	00000	00000	NOP
02308	41	00000	00000	NOP
02320	41	00000	00000	NOP
02332	49	02356	00100	B
02344	48	00000	00000	H
02356	49	00652	00000	B
02368	41	00000	00000	NOP

ERROR ROUTINES

02512	46	00736	00300	BI
02524	34	00000	00102	K
02536	39	03809	00100	WA
02548	49	02332	00000	B
02560	41	00000	00000	NOP
02572	46	00748	00300	BI
02584	34	00000	00102	K
02596	39	03849	00100	WA
02608	49	02332	00000	B
02620	46	00760	00300	BI
02632	34	00000	00102	K
02644	39	03891	00100	WA
02656	49	00000	00000	B
02668	46	00772	00300	BI
02680	34	00000	00102	K
02692	39	03943	00100	WA
02704	49	02332	00000	B

02716	41	00000	00000	NOP
02728	46	00784	00300	BI
02740	34	00000	00102	K
02752	39	03985	00100	WA
02764	49	02332	00000	B
02776	46	00796	00300	BI
02788	34	00000	00102	K
02800	39	04027	00100	WA
02812	49	02332	00000	B
02824	41	00000	00000	NOP
02836	46	01000	00300	BI
02848	34	00000	00102	K
02860	39	04065	00100	WA
02872	49	02332	00000	B
02884	46	01036	00300	BI
02896	34	00000	00102	K
02908	39	04117	00100	WA
02920	49	02332	00000	B
02932	46	01072	00300	BI
02944	34	00000	00102	K
02956	39	04169	00100	WA
02968	49	02332	00000	B
02980	41	00000	00000	NOP
02992	46	01108	00300	BI
03004	34	00000	00102	K
03016	39	04221	00100	WA
03028	49	02332	00000	B
03040	46	01120	00300	BI
03052	34	00000	00102	K
03064	39	04289	00100	WA
03076	49	02332	00000	B
03088	41	00000	00000	NOP
03100	46	01132	00300	BI
03112	34	00000	00102	K
03124	39	04357	00100	WA
03136	34	00000	00102	K
03148	39	06745	00100	WA
03160	49	02332	00000	B
03172	46	01300	00300	BI
03184	34	00000	00102	K
03196	39	04409	00100	WA
03208	34	00000	00102	K
03220	39	06745	00100	WA
03232	49	02332	00000	B
03244	41	00000	00000	NOP
03256	46	01372	00300	BI
03268	34	00000	00102	K
03280	39	04457	00100	WA
03292	49	02332	00000	B
03304	46	01384	00300	BI
03316	34	00000	00102	K

03328	39	04529	00100	WA
03340	49	02332	00000	B
03352	46	01576	00300	BI
03364	34	00000	00102	K
03376	39	04601	00100	WA
03388	49	02332	00000	B
03400	46	01588	00300	BI
03412	34	00000	00102	K
03424	39	04667	00100	WA
03436	49	02332	00000	B
03448	41	00000	00000	NOP
03460	46	01636	00300	BI
03472	34	00000	00102	K
03484	39	04733	00100	WA
03496	34	00000	00102	K
03508	39	06453	00100	WA
03520	34	00000	00102	K
03532	39	06745	00100	WA
03544	49	02332	00000	B
03556	41	00000	00000	NOP
03568	46	01648	00300	BI
03580	34	00000	00102	K
03592	39	04819	00100	WA
03604	49	03496	00000	B
03616	46	01660	00300	BI
03628	34	00000	00102	K
03640	39	04913	00100	WA
03652	49	03496	00000	B
03664	46	01972	00300	BI
03676	34	00000	00102	K
03688	39	05009	00100	WA
03700	49	02332	00000	B
03712	15	00653	00001	TDM
03724	34	00000	00102	K
03736	39	05259	00100	WA
03748	34	00000	00102	K
03760	39	05343	00100	WA
03772	34	00000	00102	K
03784	39	05459	00100	WA
03796	49	02344	00000	B

ERROR TYPEOUT DATA

03808	00	70777	27400		O	7	2	4		
03820	00	59454	14400		R	E	A	D		
03832	43	48454	3520#		H	E	C	K	#	3
03844	0#	00707	07773		O	0	W	R	H	
03856	76	00006	65949		C	O	C	7		
03868	63	45004	34845		T	E	#	8	I	E
03880	43	520#0	#0070		C	K	F	C	T	0
03892	70	77747	80000		O	7	C	A	#	
03904	56	21460	04955		O	/	O	N	O	
03916	44	49434	16356		D	I	O	T	#	
03928	59	00565	50#0#		R	O	M	6	R	
03940	00	70707	77670		E	C	C	R	E	
03952	00	00544	25920		K	#	#	7	6	
03964	45	00434	84543		7	7	2	8	6	
03976	52	0#0#0	07070		B	R	-	7	6	
03988	77	77720	00054		H	E	C	7	6	
04000	42	59205	60043		O	0	O	8	6	
04012	48	45435	20#0#		C	H	M	A	6	
04024	00	70707	77874		H	E	C	K	6	
04036	00	00544	15900		#	0	O	9	6	
04048	43	48454	3520#		D	T	M	A	6	
04060	0#	00707	07977		S	#	C	K	6	
04072	76	00005	94541		D	0	C	R	6	
04084	44	00435	20071		T	#	T	E	6	
04096	62	63005	94541		D	1	A	0	6	
04108	44	0#0#0	07071		D	2	D	N	6	
04120	70	71720	00059		O	1	A	D	6	
04132	45	41440	04352		E	0	2	A	6	
04144	00	72554	40059		E	1	2	D	6	
04156	45	41440	#0#0#0		A	0	A	D	6	
04168	70	71707	47800		D	1	0	4	6	
04180	00	59454	14400		C	K	0	8	6	
04192	43	52007	35944		R	E	1	3	6	
04204	00	59454	1440#		D	0	1	A	6	
04216	0#	00707	17079		#	1	0	1	6	
04228	76	00005	44259		6	E	1	0	6	
04240	20	45004	35200		-	F	T	0	6	
04252	41	46634	55900		A	A	R	0	6	
04264	43	41594	40059		C	E	D	0	6	
04276	45	41440	#0#0#0		E	A	D	0	6	
04288	70	71717	07800		O	1	1	8	6	
04300	00	54425	92056		M	B	R	-	6	
04312	00	43520	04146		T	C	K	A	6	
04324	63	45590	04341		R	D	R	C	6	
04336	59	44005	94541		D	#	#	E	6	
04348	44	0#0#0	07071		1	2	0	0	6	
04360	71	72700	00059		E	A	D	-	6	
04372	45	41442	04955		O	1	A	I	6	
04384	00	41594	54100		/	2	R	A	6	
04396	56	21460	#0#0#0		0	1	F	#	6	
04408	70	71727	07400		N	2	0	4	6	
04420	00	55560	07162		O	1	O	1	6	
04432	63	00435	65457		T	C	O	M	6	
04444	41	59450	30#0#		A	R	E	#	6	
04456	70	71737	67000		O	1	3	0	6	
04468	00	54425	92045		M	B	R	-	6	

04480	00	43520	04146		C	K	A	F
04492	63	45590	07162		T	E	1	S P
04504	63	00435	65457		T	R	M	
04516	41	59450	70#00		A	O	#	
04528	70	71737	77200		T	E	2	- A
04540	00	54425	92056		T	R	-	O F S P
04552	00	43520	04146		A	1	M	
04564	63	45590	07162		T	R	1	S P
04576	63	00435	65457		T	E	0	
04588	41	59450	70#00		T	R	#	
04600	70	71757	67400		A	5	6	
04612	00	54425	92045		T	M	4	- E E C E
04624	00	43520	04245		T	C	-	B
04636	46	56594	50043		A	K	M	
04648	56	54574	15945		T	O	A	R
04660	03	0#007	07175		A	M	0	1
04672	77	76000	05442		F	C	R	5
04684	59	20560	04352		O	O	O	B K
04696	00	42454	65659		F	E	F	C O M P
04708	45	00435	65457		O	C	O	
04720	41	59450	30#00		E	E	#	
04732	70	71767	27400		A	E	4	
04744	00	45216	90063		A	6	2	T F P A L
04756	47	59005	64646		G	R	Z	F M D O
04768	23	00435	65457		A	C	O	
04780	41	59450	04441		T	E	F	
04792	63	41004	65653		A	R	S	
04804	53	56666	2030#		T	O	6	
04816	00	70717	67376		L	1	3	- A C E A L
04828	00	00544	25920		L	M	R	
04840	45	00435	20041		E	T	D	
04852	46	63455	90043		F	E	O	
04864	56	54574	15945		O	M	P	
04876	23	00000	04441		T	C	A	
04888	63	41004	65653		T	K	F	
04900	53	56666	2030#		L	O	S	
04912	00	70717	67478		L	1	4	- A C E A L
04924	00	00544	25920		O	M	R	
04936	56	00435	20041		F	T	D	
04948	46	63455	90043		O	E	O	
04960	56	54574	15945		F	M	A	
04972	23	44416	34100		,	P	T	
04984	00	46565	35356		D	A	A	
04996	66	62030	70000		F	O	L	
05008	70	71797	67000		W	S	#	
05020	00	66594	96345		0	1	6	
05032	00	43520	04146		W	9	I	
05044	63	45590	05764		C	R	T	
05056	55	43480	30#		T	E	A	
					N	C	H.	

CARD COUNTER TYPEOUT DATA

05068			
05080			
05092		5945	
05104	41	44455	90043
05116	41	59440	04356
05128	64	55634	55900
05140	00	33000	£0057
05152	64	55434	80043
05164	41	59440	04356
05176	64	55634	55900
05188	00	00330	00£00
05200	43	41594	40043
05212	56	64556	34559
05224	00	59456	24563
05236	00	63560	06945
05248	59	56030	£

R	E				
C	O				
A	D	E	R	R	C
A	R	D	E	C	O
U	N	T	E	R	P
U	N	=	≠	H	C
U	N	C	H	C	O
A	R	D	E	R	C
U	N	T	E	R	O
C	A	R	D	T	C
O	A	U	N	E	R
R	T	O	S	E	T
O	O	. .	≠	Z	E
R	O	.	≠	Z	E

HEADING TYPEOUT DATA

		0071		1
05260	76	72720	04341	C A
05272	59	44005	94541	R E A U E O
05284	44	45592	05764	D E R - P D I
05296	55	43480	04445	N C H Y = P U N L
05303	53	41680	04956	L A Y . ≠ O N
05320	70	73030	£0000	O 3 . 1 O
05344	62	66007	10056	S W . 1 O
05356	55	33005	76455	N = P U N L
05368	43	48005	65553	C H . O N L
05380	68	03000	00000	Y . S W . 2 E
05392	00	62660	07200	O N = R E
05404	56	55330	05945	A D = A N D
05416	41	44004	15544	C O M P A
05428	00	43565	45741	R E O N L
05440	59	45005	65553	Y . ≠ O N = S W
05452	68	030£0	06266	3 B Y - P A
05464	00	73005	65533	S S . E T O
05476	00	42682	05741	S S . 4 S T
05488	62	62004	56356	O N = S T
05500	62	03000	00000	O P C H O D
05512	62	66007	40000	- C H O D C
05524	56	55330	06263	E L A Y E
05536	56	57004	34856	H A N G E .
05548	20	43485	60044	≠ ≠ ≠
05560	45	53416	80043	
05572	48	41554	74503	
05584	0£	0£0£0	00000	

CARD COMPARE DATA

05596	00	00000	04142	A	B
05608	43	44454	64748	G	H
05620	49	51525	35455	M	N
05632	56	57585	96263	S	T
05644	64	65666	76869	Y	Z
05656	70	71727	37475	4	5
05668	76	77787	93424	@	(
05680	04	33142	01013	+	\$
05692	21	03230	04142	A	B
05704	43	44454	64748	G	H
05716	49	51525	35455	M	N
05728	56	57585	96263	S	T
05740	64	65666	76869	Y	Z
05752	70	71727	37475	4	5
05764	76	77787	93424	@	(
05776	04	33142	01013	+	\$
05788	21	03230	04142	A	B
05800	43	44454	64748	G	H
05812	49	51525	35455	M	N
05824	56	57585	96263	S	T
05836	64	65666	76869	Y	Z
05848	70	71727	37475	4	5
05860	76	77787	93424	@	(
05872	04	33142	01013	+	\$
05884	21	03230	04142	A	B
05896	43	44454	64748	G	H
05908	49	51525	35455	M	N
05920	56	570##	70000	O	P

READ-IN CLEAR DATA

06088	00	00000	00#0#	#	#
06100	00	00000	00#00	#	
06112	00	00000	#0000		#
06124	00	000#0	00000		#
06136	00	0#000	00000		#
06148	0#	00000	0000#	#	
06160	00	000#0	#0#00	#	# # #

FIRST CHARACTER COMPARE DATA

06172	00	41424	34445	F	A	B	C	D	E
06184	46	47484	95152	G	H	I	J	K	Q
06196	53	54555	65758	L	M	N	P	V	W
06203	59	62636	46566	R	S	T	U	O	1
06220	67	68697	07172	X	Y	Z	0	7	=
06232	73	74757	67778	3	4	5	6)	*
06244	79	34240	43314	9	@	(\$	/	D	,
06256	20	10132	10323	-	+	A	B	J	K
06268	00	41424	34445	F	G	H	I	P	Q
06280	46	47484	95152	L	M	N	U	V	W
06292	53	54555	65758	R	S	T	O	1	=
06304	59	62636	46566	X	Y	Z	6)	*
06316	67	68697	07172	3	4	5	6	/	,
06328	73	74757	67778	9	@	(\$)	D	K
06340	79	34240	43314	-	+	A	B	J	P
06352	20	10132	10323	F	G	H	I	P	Q
06364	00	41424	34445	L	M	N	O	V	W
06376	46	47484	95152	R	S	T	U	V	
06388	53	54555	65758						
06400	59	62636	46566						
06412	60								

COMPARE WORKING AREA

06424		41424	34445	F	A	B	C	D	E
06436	46	47484	95152	G	H	I	J	K	Q
06448	53	54555	65758	L	M	N	P	V	W
06460	59	62636	46566	R	S	T	U	O	1
06472	67	68697	07172	X	Y	Z	0	7	=
06484	73	74757	67778	3	4	5	6)	*
06496	79	34240	43314	9	@	(\$	/	D	,
06503	20	10132	10323	-	+	A	B	J	K
06520	00	41424	34445	F	G	H	I	P	Q
06532	46	47484	95152	L	M	N	U	V	W
06544	53	54555	65758	R	S	T	O	1	=
06556	59	62636	46566	X	Y	Z	6)	*
06568	67	68697	07172	3	4	5	6	/	,
06580	73	74757	67778	9	@	(\$)	D	K
06592	79	34240	43314	-	+	A	B	J	P
06604	20	10132	10323	F	G	H	I	P	Q
06616	00	41424	34445	L	M	N	U	V	W
06628	46	47484	95152	R	S	T	O	1	=
06640	53	54555	65758	X	Y	Z	6)	*
06652	59	62636	46566	3	4	5	6	/	,
06664	67	68697	07172	9	@	(\$)	D	K
06676	73	74757	67778	-	+	A	B	J	P
06688	79	34240	43314	F	G	H	I	P	Q
06700	20	10132	10323	L	M	N	O	V	≠
06712	00	41424	34445						
06724	46	47484	95152						
06736	53	54555	6570						
	60	7000	00000						

