CONTROL DATA
CORPORATION

CONTROL DATA®
1700 SYSTEM
MAINTENANCE MONITOR (SMM17)

VOLUME 1
REFERENCE MANUAL

RECORD of REVISIONS

REVISION	NOTES
01	Original Printing, preliminary edition.
(5-13-66)	
02	Publications Change Order 14307. Reprint with revision which obsoletes all previous editions.
(8-8-66)	Tests were updated and the following new tests were added: 1711/1712 Teletype, 1729 Card Reader,
	1731 Magnetic Tape, 1706/1716 Buffered Data Channel and Coupling, Random Protect, 1700 SMM
	Edit Routine, and Enter Program.
A	Manual released. Publication Change Order 16368. The following new tests are added: 0B (1718
(5-1-67)	Satellite Coupler Test), 0C (1742 Line Printer Test), and 3D (Enter Program). Other tests were
	extensively revised and updated. This edition obsoletes all previous editions.
B	Publication Change Order 17146. To revise existing tests and add new tests. Introduction: page 5
(9-14-67)	revised. Description: pages 7, 12, 15, 18, 25, 26, 27, 30 and 35 revised. Pages 30-a and 30-b
	added. Tests: pages 90-1, 90-2, 100-7, 100-8, 100-10, 101-2, 101-7, 202-1, 202-7, 205-2, 206-6
	thru 206-10, 207-3, 208-2 and 208-6 revised. Page 100-8a added. Tests sections: 102, 201, 203,
	212, 213 and 214 added. Sections 102 Rev A, 201 Rev A and 203 Rev A removed.
C	Publications Change Order 18929. To add 1728 Card Reader/Punch test, No. D.
(2-28-68)	
D	Publications Change Order 19818, to make miscellaneous publication corrections. Pages 37,
(6-11-68)	100-2, 100-18, 101-9, 102-7, 200-10, 201-6, 202-9, 203-7, 204-1, 204-12, 205-14, 206-9, 206-10
	207-4, 208-21, 210-4, 210-6, 211-13, and 215-23 revised. Pages 207-5 and 212-24 added.
E	Manual Revised, Engineering Change Order 21307, publications change only. Information included
(1-6-69)	through Edition 2.1. Pages 35, 90-1, 90-2, 90-6, 101-10, and 208-1 thru 208-21 revised; pages
	30-c through 30-f, 51 through 60, 103, 216, 217, 218, 219, 220, 221, 222 and red tab dividers
	added. Manual divided into two volumes.
F	Manual revised, Engineering Change Order 21883. This manual is complete through Edition 2.1.
(12-15-69)	
G	Manual revised. New tests are added and editorial corrections made. This manual is complete
(2-15-70)	through Edition 2.2.
H	Manuals revised. This publication is complete through Ed. 2. 3. All previous editions are obsolete.
(12-15-70)	
J	Manuals revised. New tests are added and minor corrections are made. This publication is
(2-5-73)	complete through Ed. 3. 0.
K	Manuals revised. Tests are added, deleted, and corrected.
(9-20-73	
L	Manuals revised. Tests are added, deleted, and corrected. This publication is complete through
(2-1-74)	Edition 3. 1.

Address comments concerning this manual to:

Control Data Corporation
Technical Publications Department
4201 North Lexington Ave.
Arden Hills, Minnesota 55112

or use Comment Sheet in the back of this manual.

Pub. No. 60182000

Printed in United States of America

PREFACE

MANUAL STRUCTURE

This manual is intended to serve as a reference aid for field and checkout personnel involved in the running of the CONTROL DATA® 1700 System Maintenance Monitor (SMM17). It consists of two sections:

SMM17 DESCRIPTION

A detailed description of the operation and use of the monitor, instructions for the operator, restrictions and necessary parameters. An asterisk (*) on the left of the page will highlight operator tasks. Supplements are included in the back of this section.

TESTS

Detailed test descriptions complete the three volume reference manual.

CONTENTS

Optical Readers

VOLUME 3

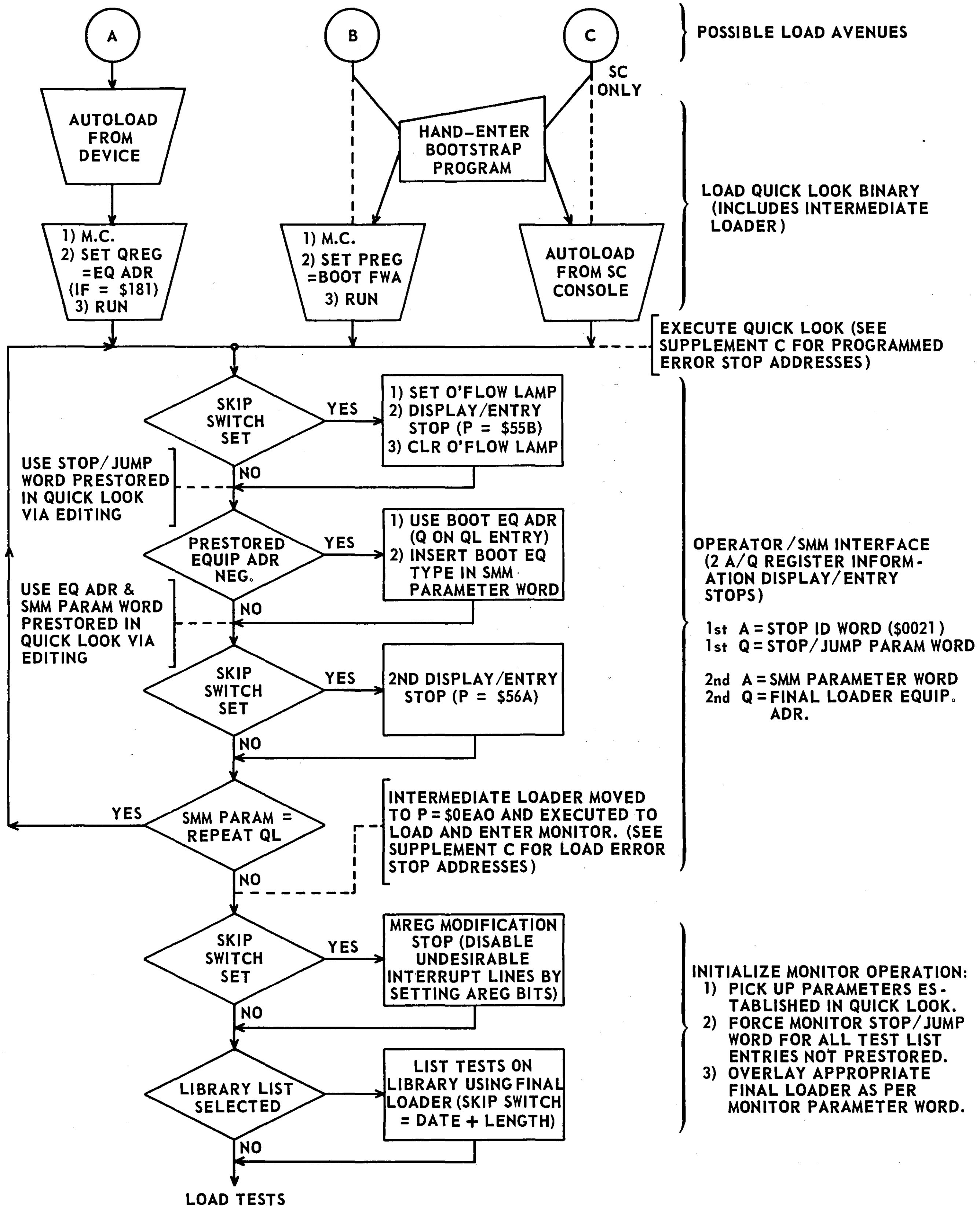
Communication Equipment

Analog/Digital

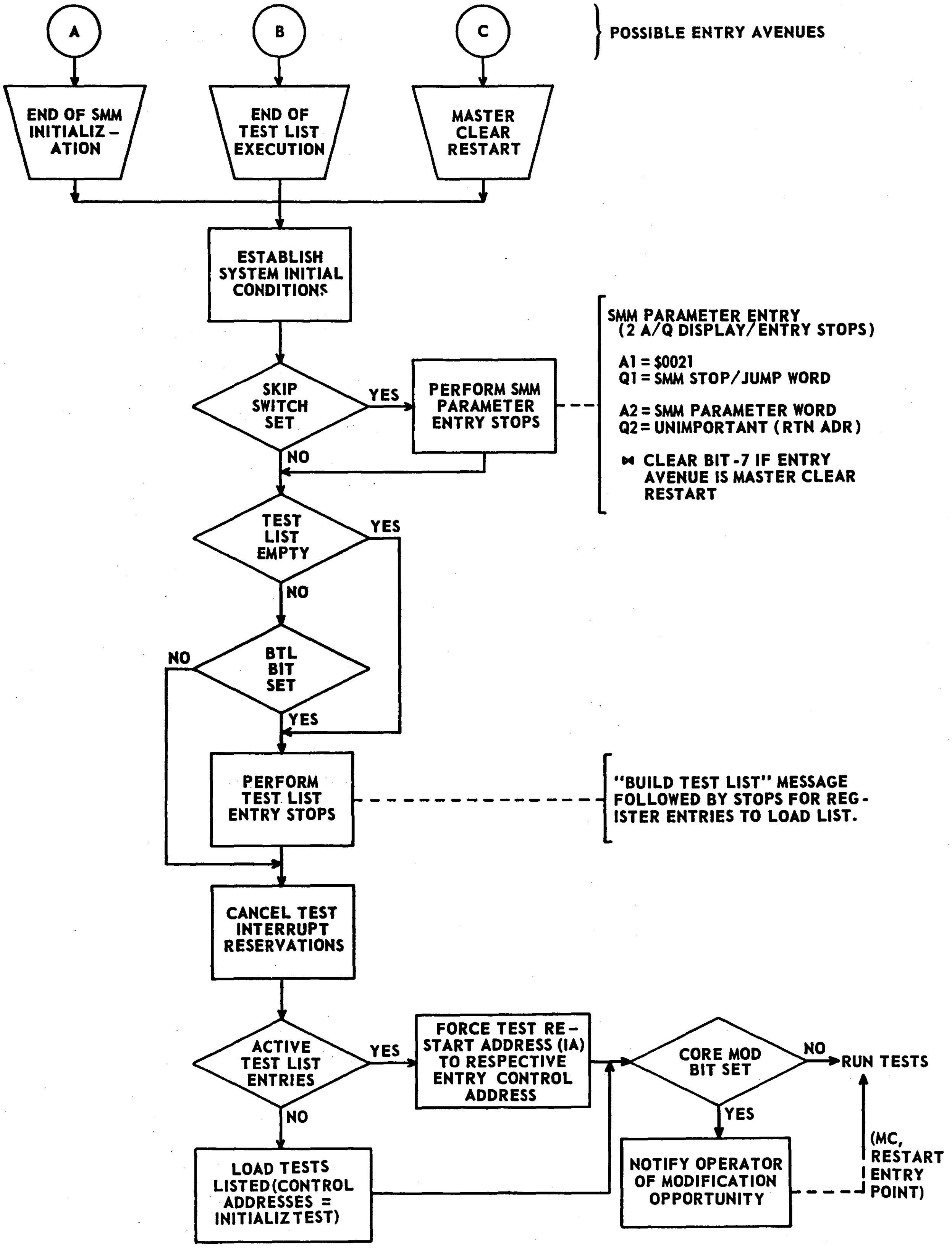
Miscellaneous

QSE TESTS

LOADING AND INITIALIZING SMM17


A
B
C
POSSIBLE LOAD AVENUES
SC ONLY
AUTOLOAD FROM DEVICE
HAND-ENTER BOOTSTRAP PROGRAM
LOAD QUICK LOOK BINARY (INCLUDES INTERMEDIATE LOADER)
1) M.C. 2) SET QREG =EQ ADR (IF = $181) 3) RUN
1) M.C. 2) SET PREG =BOOT FWA 3) RUN
AUTOLOAD FROM SC CONSOLE
EXECUTE QUICK LOOK (SEE SUPPLEMENT C FOR PROGRAMMED ERROR STOP ADDRESSES)
SKIP SWITCH SET
YES
1) SET O'FLOW LAMP 2) DISPLAY/ENTRY STOP (P = $55B) 3) CLR O'FLOW LAMP
NO
USE STOP/JUMP WORD PRESTORED IN QUICK LOOK VIA EDITING
PRESTORED EQUIP ADR NEG.
YES
1) USE BOOT EQ ADR (Q ON QL ENTRY) 2) INSERT BOOT EQ TYPE IN SMM PARAMETER WORD
NO
USE EQ ADR & SMM PARAM WORD PRESTORED IN QUICK LOOK VIA EDITING
OPERATOR/SMM INTERFACE (2 A/Q REGISTER INFORMATION DISPLAY/ENTRY STOPS)
1st A = STOP ID WORD ($0021)
1st Q = STOP/JUMP PARAM WORD
2nd A = SMM PARAMETER WORD
2nd Q = FINAL LOADER EQUIP. ADR.
SKIP SWITCH SET
YES
2ND DISPLAY/ENTRY STOP (P = $56A)
NO
YES
SMM PARAM = REPEAT QL
INTERMEDIATE LOADER MOVED TO P = $0EA0 AND EXECUTED TO LOAD AND ENTER MONITOR. (SEE SUPPLEMENT C FOR LOAD ERROR STOP ADDRESSES)
NO
SKIP SWITCH SET
YES
MREG MODIFICATION STOP (DISABLE UNDESIRABLE INTERRUPT LINES BY SETTING AREG BITS)
NO
INITIALIZE MONITOR OPERATION:
1) PICK UP PARAMETERS ESTABLISHED IN QUICK LOOK.
2) FORCE MONITOR STOP/JUMP WORD FOR ALL TEST LIST ENTRIES NOT PRESTORED.
3) OVERLAY APPROPRIATE FINAL LOADER AS PER MONITOR PARAMETER WORD.
LIBRARY LIST SELECTED
LIST TESTS ON LIBRARY USING FINAL LOADER (SKIP SWITCH = DATE + LENGTH)
NO
LOAD TESTS

LOADING TESTS


A
B
C
POSSIBLE ENTRY AVENUES
END OF SMM INITIALIZ - ATION
END OF TEST LIST EXECUTION
MASTER CLEAR RESTART
ESTABLISH SYSTEM INITIAL CONDITIONS
SKIP SWITCH SET
YES
NO
PERFORM SMM PARAMETER ENTRY STOPS
TEST LIST EMPTY
YES
NO
BTL BIT SET
NO
YES
PERFORM TEST LIST ENTRY STOPS
CANCEL TEST INTERRUPT RESERVATIONS
ACTIVE TEST LIST ENTRIES
YES
NO
FORCE TEST RE - START ADDRESS (IA) TO RESPECTIVE ENTRY CONTROL ADDRESS
LOAD TESTS LISTED(CONTROL ADDRESSES = INITIALIZTEST)
CORE MOD BIT SET
NO
YES
RUN TESTS
NOTIFY OPERATOR OF MODIFICATION OPPORTUNITY
(MC, RESTART ENTRY POINT)

SMM PARAMETER ENTRY
(2 A/Q DISPLAY/ENTRY STOPS)

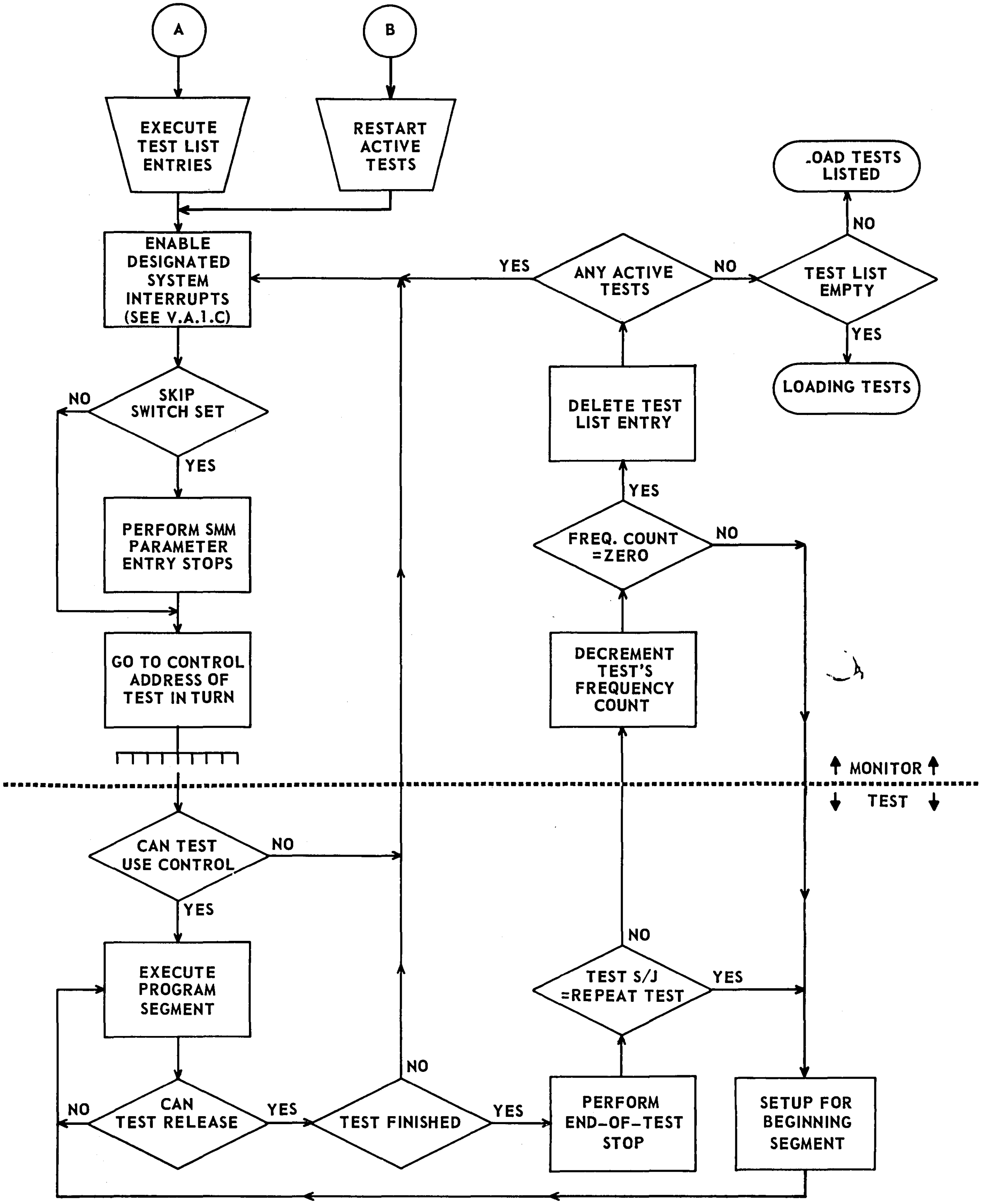
A1 = $0021
Q1 = SMM STOP/JUMP WORD

A2 = SMM PARAMETER WORD
Q2 = UNIMPORTANT (RTN ADR)

⋈ CLEAR BIT-7 IF ENTRY AVENUE IS MASTER CLEAR RESTART

"BUILD TEST LIST" MESSAGE FOLLOWED BY STOPS FOR REGISTER ENTRIES TO LOAD LIST.

RUNNING TESTS



SMM17 DESCRIPTION

I. SMM DEFINITION

The 1700 System Maintenance Monitor (SMM17) is a set of programs designed to be run as an aid to maintenance and checkout of the CONTROL DATA® 1700 series computer system. This set of programs is composed of a monitor and a collection of test or diagnostic programs. The monitor is an executive routine which controls the running of the tests. Each test checks or diagnoses a particular peripheral equipment or some part of the main (1704/1714/1774/1784) computer.

II. LOADER DEFINITION

Getting the monitor and tests read in requires three load operations. First, a Quick Look Command Test is loaded by an autoload or a hand-entered Bootstrap. After executing the Quick Look Command Test, an Intermediate Loader within Quick Look loads the monitor. Finally, the monitor's Final Loader will load the tests. All references to these loaders will be made by the underlined terms above.

III. LOADING SMM

A. Quick Look, a basic instruction checkout program, is first loaded and executed by either of two methods: Hand-Entered Bootstrap or Autoload (see III. A. 1 and 2). Quick Look stops on instruction failure. The P register contents for these programmed error stops are tabulated in Supplement C.

1. Hand-Entered Bootstrap

   a. Master clear.

   b. If on an SC-1700 System Controller, press AUTOLOAD PROTECT button.

   c. Press P register select button.

   d. Set P register to $XFE0 (X = number of highest bank).

   e. Set ENTER/SWEEP to ENTER.

   f. Press X register select button.

   g. Clear X register with clear button.

   h. Enter word of bootstrap in X (see Supplement A for bootstrap program).

i. Push RUN/STEP switch to STEP.

j. Repeat steps g, h, and i until bootstrap has been entered.

k. Return ENTER/SWEEP switch to center position.

l. Master clear.

m. Set SELECTIVE STOP and SKIP switches.

n. Make input device READY.

o. Press P register select button.

p. Set P register to $XFE0 (first word address of bootstrap).

q. Place RUN/STEP switch in RUN to execute the bootstrap, loading and entering Quick Look.

2. Autoload

a. Device autoload.

1) Master clear.

2) Make input device READY.

3) Press device AUTOLOAD button.

4) Set SELECTIVE STOP and SKIP switches.

5) If input device equipment address for direct input (Director Status 1) is $0181 or edit prestored value, go to step 6. Otherwise, enter value in Q register.

6) Place RUN/STEP switch in RUN to execute Quick Look.

b. Console Autoload (SC-1700 System Controller only)

1) If bootstrap program is not in core, do hand entry described above (steps a. through j.).

2) Make input device READY.

3) Set SELECTIVE STOP and SKIP switches.

4) Press AUTOLOAD button to execute the bootstrap, loading and entering Quick Look.

B. Following Quick Look execution there are two programmed stops for register display/entry of equipment and operation information to be used by monitor; Quick look will terminate with A register = $0021 (Monitor ID) and Q register = $020X (System STJP Parameter). Any other information at this stop is an error. See supplement C for the P address meaning.

1. First Stop

a. A register = stop series identification word, $0021 (see V. A.)

b. Q register = system Stop/Jump parameter word (see V. B.).

c. Make desired changes, if any.

d. Place RUN/STEP switch in RUN.

2. Second Stop

a. A register = SMM parameter word (set bits 2 and 3 for the appropriate memory speed of processor).

b. Q register = SMM library device equipment address for final loader.

c. Make desired changes, if any.

d. Clear SELECTIVE SKIP switch. (Omit this step if an interrupt line is to be disabled.)

e. Place RUN/STEP switch in RUN.

C. The intermediate loader in the Quick Look binary program loads and enters monitor, making accessible the information words above. (If the SKIP switch is left up through this load, a third stop will occur allowing the operator to permanently disable system interrupt lines by setting corresponding A register bits. After this point, the SKIP switch will trigger an SMM parameter display/ entry stop series.)

IV. LOADING AND EXECUTING TESTS

A. After monitor initialization and heading typeout (assuming message device availability) monitor will type Build Test List and begin a series of programmed stops to allow the operator to enter test numbers into a list for subsequent loading of tests.

1. Test Entry

a. A register format = TTRR (T and R are 4 bit bytes, or Hex digits)
TT = Test Number (i.e. Command Test = 01, Memory Test = 02, etc.)
RR = Frequency (automatic test repeat count, at least 01)

b. Q register entry = Equipment address of tested peripheral, if any. (Zero entry uses value prestored in test on the library. See Test 3E description, EDT.)

2. List Stringer (forces load termination). Those loaded will execute; entries after stringer will be loaded when they have finished.

a. A register = $0001

b. Q register = unimportant

3. Set Test Load Address (List entries following this entry will be loaded starting at this memory address)

   a. A register = $FF00

   b. Q register = Desired load address

   c. Place RUN/STEP switch in RUN.

   d. Repeat steps indicated in IV.A.1.

B. Monitor then loads from the library according to this list, enters each test one-at-a-time for test initialization and heading typeout, re-enters each for test parameter entry (see particular test description for parameter definition), then continues this program control multiplexing to the completion of each list entry, resulting in its deletion from the list. When the list is empty, monitor re-enters its Build Test List phase.

V. SMM/OPERATOR INTERFACE

A. PROGRAMMED INFORMATION STOPS

Four types of stops for information display/entry during system execution may occur: Parameter Entry, End of Test Section, End of Test, and Error Stops. These stops are handled by a monitor subroutine. (Special test programming may result in other peculiar stop possibilities. Such stops will be detailed in the deviating test's description.) The A register contents on the first stop of a series of any type identifies the test number, series stop count, and the type of stop.

1. First Stop of Series, A1/Q1 (Overflow lamp lit for first stop)

   a. A register = XXYZ (Hex digits) Stop ID word.

```
        where XX = Stop initiator's test number (monitor = 00)
               Y = Number of stops in the series (pairs of A and Q)
                   to include this stop
               Z = Type of stop:
                   1 = Parameter
                   2 = End of Section
                   4 = End of Test
                   8 = Error
```

   b. Q register = User Stop/Jump word (open to modification) (see V.B.)

   c. M register = System Interrupt-Enable word (interrupt lines may be disabled by clearing corresponding bits)

2. Successive Stop Information

a. Parameter stop series A2/Q2 through AY/QY = parameter information (see write-up on particular user, V.C. if SMM)

b. End-of-Section A2 = SS00, where SS = section number

c. End-of-Test A2 = test execution pass count

d. Error Stop A2 = SSEE
   where SS = Test section number (00 for monitor errors)
   EE = Error code (RE: particular test description or Instant SMM Manual)

Q2 = Program address for returning to user after stop series completed (program listing reference).

B. STOP/JUMP PARAMETER

This word is the software equivalent of hardware STOP and SKIP switches. A list of Stop/Jump words is generated during SMM initialization with an entry for SMM and each possible test list entry. The SMM system value, designated during Quick Look execution, is forced for all test entries not prestored in the SMM program via editing. Any information stop series (see V.A.) will reference the stop initiator's entry only, but all entries may be forced to a given value (Stop Q1) by clearing the ID word (Stop A1) before continuing to the next stop of the series (A2/Q2).

The test Stop/Jump parameter affects only the currently active test. However, the monitor STJP (system STJP) is logically OR'ed with each of the test Stop/Jump parameters when making the decision if a stop should occur. It is important that the operator must tell the system and each test under what conditions it is to stop or jump.

Stop/Jump Bit Assignment: (*)

Bit 0	= Stop to enter test parameters	(Stop Type 1)
Bit 1	= Stop at end of test section	(Stop Type 2)
Bit 2	= Stop at end of test	(Stop Type 4)
Bit 3	= Stop on error	(Stop Type 8)
Bit 4	= Repeat condition (last test exercise)	(Jump Type 1)
Bit 5	= Repeat (current) test section	(Jump Type 2)
Bit 6	= Repeat test	(Jump Type 4)
Bit 8	= Omit typeouts (may be used in conjunction with one of the jumps for scoping purposes)	

*Unused bits may be assigned at test level

Bit 9 = Bias displayed return address of stop initiator (Stop Q2 of all but parameter entry stop series)

1 = Use actual core address

0 = Use listing address (relative to test FWA)

Bit 10 = Re-enter test parameters (bit 0 must also be set)

A more detailed explanation of the Stop/Jump parameter follows:

Bit 0 - Stop 1 (stop to enter test parameters)

Setting bit 0 will cause a stop to occur during each test's initialization and also when a test is restarted from its initial address. The stop allows the test's prestored parameters to be changed.

Bit 1 - Stop 2 (stop at end of section)

If bit 1 is set, a stop will occur after each test section selected.

Bit 2 - Stop 4 (stop at end of test)

Setting bit 2 will cause a stop to occur after each complete pass of the test.

Bit 3 - Stop 8 (stop on error)

If bit 3 is set, a stop will occur when a test or the monitor encounters an error. If no teletype is attached to the system, this bit will be set by the monitor initialization.

Bit 4 - (Repeat conditions)

Normally, bit 4 would be set only after an error stop. The failing loop would then be repeated with the same conditions until the bit is cleared.

Bit 5 - (Repeat section)

Setting bit 5 will cause the current section to be repeated until the bit is cleared.

Bit 6 - (Repeat test)

If bit 6 is set, the tests will be repeated until the bit is cleared. The selected number of passes (frequency) will have expired (see IV).

Bit 7 - (Not used)

Bit 8 - (Omit typeouts)

If bit 8 is set, no typeouts will occur. Not only will end of test and error typeouts be eliminated, but the tests messages to the operator will also be omitted. If no teletype is attached to the system, the bit will be set by the monitor initialization.

Bit 9 - (Display memory return address in all but parameter stops)

The operator is given the test address to which control will return after the stop series (Q2). If bit 9 is set, the address furnished will be the actual memory address. If cleared, the address furnished will be the address in the test listing, i.e., the program address relative to its load address.

Bit 10 - (Re-enter test parameters)

If bit 10 and bit 0 are set, another parameter stop is forced.

Bits 11-15 - (Test jumps)

See the individual test writeups for descriptions, where used.

At any ID stop, the STJP may be altered for that active test only. In the event of a monitor ID stop, the system STJP is displayed and alterations will affect all tests. Also at any ID stop, the mask register (normally all ones) may be selected and any bit may be permanently de-selected.

C. SMM PARAMETER

This word describes operator desired system functions and operating modes to the monitor.

1. SMM Parameter Bit Assignment

*Bit 0 = List library contents. Set the SKIP switch during listing to include revision date MO/DA/Y and program length information MC - Restart to abort.

*1 = Repeat Quick Look

2-3 = Memory Speed Selection -

Bits	3	2	CPU
	0	0	1704/14
	0	1	1774
	1	0	1784 (900 nanoseconds)
	1	1	1784 (600 nanoseconds)

*These bits must be set during Quick Look's SMM parameter entry stop to be effective.

4 = Load tests at address multiples of $100.

5 = Type messages in Non-Interrupt mode. (Without bit 5 set, typeouts are driven by data interrupts and testing is resumed between characters. Setting bit 5 forces continuous typeout with testing inhibited until the message is complete.)

*6 = Select teletype input mode (may only be selected at end of Quick Look but may be de-selected at any time.)

7 = Stop to build test list (see IV).

*8 = Select MBS (special monitor based subroutine section used by tests so designated).

9 = Ignore edited program modification on test load. (Load test without prestored correction.)

10 = Not used.

11 = Test Correction Stop. Stop for pre-execution modification of test(s) loaded (see Monitor Error Code 20). This bit tested after test loading and after test restart (see VI. A).

*12-15 = Loader type: from which TESTS will be loaded

Bits	15	14	13	12	Type
	0	0	0	1	1 = Paper Tape (1712, 1713, 1721, 1777)
	0	0	1	0	2 = Cards (1726, 1728, 1729-2, 1729-3)
	0	0	1	1	3 = Mag Tape (1731, 1732-608/609, 1732-2-6x8/6x9)
	0	1	0	0	4 = Disk Pack (1738/853,4 - FA706/ 853,4, 1733)
	0	1	0	1	5 = CDD (1739/1733-2)

The loaders are pre-selected to the bootstrap input type but may be changed to any other type at the end of Quick Look only. The contents of Q register must be changed to the correct equipment address at this time, also.

2. Q = Loader equipment address.

This may be changed at anytime to another address of the same loader type. See monitor parameter description for other uses.

*These bits must be set during Quick Look's SMM parameter entry stop to be effective.

D. SYSTEM MESSAGES

Program headings, key messages, and stop-information (see V.A.) will be recorded if a message device is available to the system and bit 8 of the appropriate Stop/Jump word is clear. Test messages are described in test write-ups. Stop-information messages and monitor messages will be obvious.

VI. SYSTEM USAGE

A. TEST RESTART

A particular sequence of system events causing an error condition may be recreated by a Master Clear Restart of the system. (Only those tests active in core at time of MC Restart will restart. Those having run to completion must be re-loaded.) The process is as follows:

1. Multiple Tests Loaded

   a. Master clear.

   b. If bit 7 of the SMM parameter word Build Test List is clear, go to e.

   c. Set SKIP switch to flag Monitor Parameter stop.

   d. Clear bit 7 and SKIP switch.

   e. Run.

Single Test Loaded

Any particular test may be re-started (after its initialization) during its program control cycle, if it is not executing the monitor's message subroutine, as follows:

a. Master clear.

b. Set P register = test IA (load address).

c. Run.

B. TTY INPUT PACKAGE SELECTION

Bit 6 in the SMM parameter word (stop 2 of A) of monitor will select the TTY input package. This can only be selected when loading monitor. Once selected, the package will remain selected until deselected (bit removed from SMM parameter word). Deselected package cannot be reselected again unless monitor is reloaded.

OPERATION

Every stop of A and Q is typed A slash Q followed by a question mark and bell. Example: $0021/$0201? If no changes are needed, a CR (carriage return) continues operation.

Alteration to the offered parameters may be done in any of the following methods.

The first character typed in must be:

$ designating a maximum of four hexadecimal characters to follow right justified. Example: $21 CR = 0021_{16}.

B designating bit positions to be set in any order. Example: B, 0, 5 CR = 0021_{16}

- Any legal decimal number without a prefix will be interpreted as a hex number. Example: 33 CR = 0021_{16}.

P Set P register and execute. Example: P$1000 CR = Jump to location 1000 and execute.

All other prefixes will result in an error; the TTY will respond with ER Bell. The parameters may still be changed starting with A.

If a mistake is made while typing in the parameters, a Rubout CR will nullify your line of type and the TTY will respond with ? bell.

The operator need only change the displayed register desired without copying the other.

If the A register is to be changed and not Q, type in the change and terminate with a CR .

If Q register is to be changed, type in / your parameter CR .

If both are acceptable, type in CR .

Examples

BUILD TEST LIST	Typeout
$0201/$0000? bell	Test 1 change?
$301/$C1 CR	Yes Test #03 eq 00C1
$0101/$0000? bell	Test 2 change?
501/ RO CR	Yes want Test #05 mistake
?$501/B, 0, 4, 9 RO CR	Test #05 equip adrs 0091 mistake
?$501/B, 0, 7, 4 CR	Test #05 equip adrs 0091
$0000/$0000? bell	Test 3 change?
257/0 CR	Test #01 (257_{10}=101_{16}) adrs O
$0000/$0000? bell	Test 4 change?
H ER? bell	Illegal char identified error
0 CR	Terminate library list load tests

C. WORST CASE SETUPS

Certain combinations of tests being run together are more effective at exercising the equipment than a test being run alone. Running several tests together may reveal equipment conflicts and critical timing considerations. This procedure may be helpful in simulating another system (i.e., MSOS) which detects a failure, but is not adapted for error isolation. Some examples follow:

1. Running one or more copies of COM (Command Test) with a peripheral test (non run alone) is an excellent method of introducing a random start/stop motion. Each additional copy of COM increases the delay between motion operations. The method has been especially effective with PTR Paper Tape Reader Test.

2. Disk Pack Test DP1 may be loaded at various addresses (see IV.3) for checking direct storage access (DSA) addressing in various areas of core. Only the surface test section should be run during this check. Component isolation may require hand-entry of a small, more specific, addressing exercise routine once the failing area has been located.

3. When the operating system (MSOS) fails without a clear indication of what happened, running several SMM tests together can, in some cases, simulate the failing conditions of the operating system. To determine what tests to run, note what Input/Output operations were in progress when the operating system failed. Then, run the tests on those equipments that were active in the operating system failure. Also, the sections to be run should be selected to simulate the operating system failure. For instance, an assembly from a source tape fails, the tests to run would be a Magnetic Tape Test and the Disk Pack Test (DP1 with Sections 3 and 7 selected).

D. LOAD AND EXECUTION AUTOMATION

Tailoring a particular system's SMM library according to equipments available can result in a relatively automatic system load and execution (except for the bootstrap). Peripheral equipment addresses as well as test parameter words are easily installed in the library programs via Test 3E, the system editing program (Monitor's Parameter word and system Stop/Jump word are contained in Quick Look, Test FF). Operator understanding of SKIP switch use during system load and initialization (see first flow chart, this section), and familiarity with the SMM/operator interface mechanisms (display/entry stop scheme, Stop/Jump and SMM parameter bit significance) will then control efficient system usage.

VII. MONITOR ERROR CODES

The stop series identification word (A1) = $00X8 for all monitor errors (X = 2 or 3). One of the following codes will appear in the A register on the second stop of the series (A2) and have the described significance:

Code	Description
01	Memory parity error. A3=interrupted program address.
02	Protect fault. A3=interrupted program address.
03	Monitor message. Clear computer PROTECT switch(es).
04	Unrequested interrupt. A3=decimal line number.

* I/O errors reported during lib list include stop A4/Q4 = 1st/3rd record words. (A4-block ID. Q4=program address of block data if ID=RBDBLK, $4050.)

Code	Description
05	Interrupt line request conflict. A3=requestor's IA Q3=assigned interrupt processor's address. (M.C.from error stop, restart at test IA for recovery.)
06	Teletype rejected data after data interrupt.
07	MBS could not return to test after interrupt within the time specified by the test.
10	Load area insufficient for test. Retried after tests in core ran. Aborted if none. A3=test/PREG, Q3=assigned IA.
11	Loader encountered illegal RBD block ID. A3=ID word. Q3=code of load error, if any. Load aborted.
12	No RBD transfer block found. Load aborted.
13	Check sum error (card or paper tape). Attempt reload.
14	Paper tape reader alarm. A3=status. Attempt reload.
15	Card reader alarm. A3=status. Attempt reload.
16	Improper card reader EOP. A3=column count. Try reload.
17	Load tape parity error. A3=status. Q3=times error recurred (load retry aborted on 8th, successful if less).
18	Disk pack alarm. A3=status. Q3=times error recurred (load retry aborted on 50th, successful if less).
19	Cartridge disk alarm. A3=status. Q3=times error recurred (load retry aborted on 50th, successful if less).
1A	Overlay loading error. If MBS overlay, reload system.
20	Monitor message. Pre-execution test modification stop. MC, make changes in core where test(s) loaded. MC, run to restart.
Non-Std Code	Power fail interrupt - system halts with the following register contents: A=0018, Q=007F. Restore power if required, MC and run.

*I/O errors reported during lib list include stop A4/Q4 = 1st/3rd record words. (A4-block ID. Q4=program address of block data if ID=RBDBLK, $4050.)

VIII. GENERAL OPERATING INSTRUCTIONS

A. There is no difference at all in initializing SMM, calling of tests, and loading of tests from any of the supported input media. If the TTY input driver is selected, each of its operations are directly equivalent to an operator's panel operation. See TTYIN.

1. Master clear system.

2. Enter respective bootstrap or autoload 1738 or 1739. Suggested address if at P = XFEO, however, it will execute correctly at any address above OFEO. (Bootstraps are located in supplement A.)

3. Master clear. Set P = bootstrap, first word address, set SELECTIVE SKIP, STOP switches and run. In the case of mass storage, autoload, master clear, and run.

   a. Bootstrap now is loading Quick Look. It will terminate when it sees the first full word of zeros, at which time it will jump to the beginning of Quick Look and execute.

   b. Quick Look will terminate with the A register = 0021 (Monitor ID) and Q = 020X (system Stop/Jump parameter). Any other stop is an error. (See Supplement C of the SMM Reference Manual for the P address meaning of the instruction that failed.) If Quick Look encounters an error, it will loop on the failing sequence until the error is corrected, usually about a 6 instruction loop.

   c. From this time on, all displays are in the following format:

      1) A1 = ID Q1 = STJP. Q2 = Return Address

      2) ID = TTNS where:

         TT = test number making this call (monitor = 00)
         N = number of stops (pairs of A and Q's) including this one to the completion of the call
         S = Stop type
         1 = parameters, 2 = end of section, 4 = end of test
         8 = error

      3) STJP = the Stop/Jump parameter affecting only this test. However, the monitor STJP (system STJP is logically OR'ed with each of the test Stop/Jump parameters when making the decision if a stop should occur).

The operator must tell the system and each test under what conditions it is to stop, and under what conditions it is to jump. Also, the lower four bits of the STJP is equivalent in meaning to the lower four bits of the ID word.

STJP Bit Assignments:

0 =	Stop to enter parameters	(Stop Type 1)
1 =	Stop at end of test section	(Stop Type 2)
2 =	Stop at end of test	(Stop Type 4)
3 =	Stop on error	(Stop Type 8)
4 =	Repeat conditions	(Jump Type 1)
5 =	Repeat section	(Jump Type 2)
6 =	Repeat test	(Jump Type 4)
7 =	Not used	

8 = Omit typeouts (may be used in conjunction with one of the jumps for scoping purposes).

9 = Bias return address display to actual memory addresses rather than listing addresses.

10 = Re-enter test parameters

11-15 = Test defined

At any ID stop, the STJP may be altered for that test only (in the case of a monitor stop, this STJP affects all tests equally).

4) When at any ID stop, the Mask register may be selected (normally all ones) and any bit may be permanently de-selected at this time.

4. RUN, CPU will stop with: A = System Parameter
Q = Loading Equipment Address

a. A = System parameter bit assignments

Bit

0 = Type out library list from selected loader and equipment address (available only at end of Quick Look)

1 = Repeat Quick Look (available only at end of Quick Look)

2 =	Memory speed selection	0 = 1704/14	1 = 1774	0 = 1784	1 = 1784
3 =		0	0	1	1

4 = Load tests at even hex 100's

5 = Type out in non-interrupt mode

6 = Select teletype input mode (may only be selected at end of Quick Look but may be de-selected at any time)

7 = One, build test list. Equal zero, execute prestored test list

8 = Select MBS package (may only be selected at end of Quick Look, but may be de-selected at any time)

9 = Ignore edited program modifications, while loading

10 = Not used

11 = Stop for program modification

12

13 = Loader type: from which tests will be loaded

14 1 = Paper Tape (1712, 1713, 1721, 1777)

15 2 = Cards (1726, 1728, 1729-2, 1729-3)

3 = Mag. Tape (1731, 1732-608/609, 1732-2-6x8/6x9)

4 = 85X Disk (1738, 1733)

5 = CDD (1739, 1733-2)

The loaders are pre-selected to the bootstrap input type but may be changed to any other type at the end of Quick Look only. The contents of Q register may be changed to the correct equipment address at this time also.

b. Q = Loader equipment address

5. Run

a. Monitor will now be loaded.

b. If the SKIP switch is set, another stop will occur to allow the operator to de-select selected interrupt lines by setting the respective bits in the A register and running again.

c. The correct loader type will now be loaded.

d. TTY will typeout SMM 17 VERS. 3.x.x. CP2F

1) Vers. 3.x.x is the version number of monitor you are using.

2) CP2F is a formed number that allows the user to determine the type configuration this module of the diagnostic package is valid for. Consider the number by bit positions. Where bit:

0 = 1.1 microseconds memory speed

1 = 1.5 microseconds memory speed

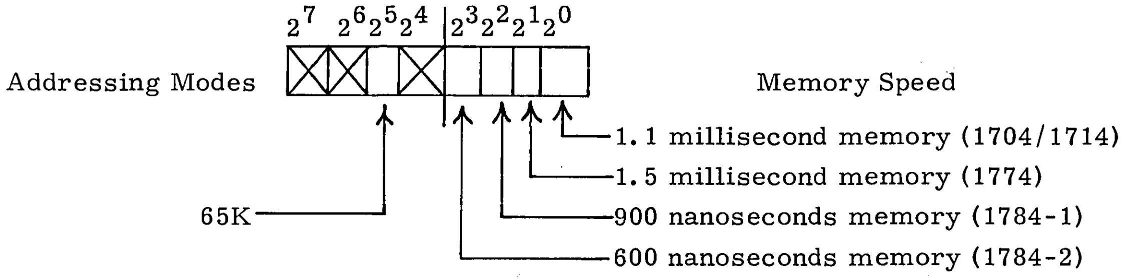
2 = 900 nanoseconds memory speed

3 = 600 nanoseconds memory speed

4 = Not used

5 = 65K mode O.K.

A 2-hexadecimal digit derived number accurately reflects the acceptable machine types and memory configurations. This number will be composed as follows:



2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0
Addressing Modes
Memory Speed
65K
1.1 millisecond memory (1704/1714)
1.5 millisecond memory (1774)
900 nanoseconds memory (1784-1)
600 nanoseconds memory (1784-2)

Example: CP01 = 32K only 1704 or 1714 only
CP2D = 65K acceptable 1704, 1714, 1784-1 or 1784-2
but not a 1774

This 2-digit number must appear in the typed out test heading and in the respective documentation of the SMM17 Reference Manual.

a. Build Test List

From this point on, all entries may be made by one of two ways:

1) From the TTY (see TTY description)

2) From the operator's panel

6. Enter the test number in A register in the following format:
TTFF.
Where TT = test number and FF = frequency count (repetition number).
Example 0105_{16} = command test 01 to be repeated five times.

a. Enter the equipment address of the device under test in Q.

b. The format is usually that bit configuration that will accomplish a status 1 from the device. Since there are some exceptions, check in the appropriate test documentation for deviations.

c. Run.

7. To multiplex tests continue entering test numbers in A and equipment addresses in Q.

8. To terminate test list enter 0 in A and 0 in Q and run.

9. To create strings of tests enter tests per step 7 but instead of step 8 enter A = 0001 and Q = 0000 and repeat step 5 terminating list per step 8.

10. To Request a test to load at a specific address enter A = FF00 and Q = ADDRESS.
    Run.
    Enter A = TTFF and Q = equipment address (step 6 or 7). Continue this process if using step 7 until the list has been entered.

11. Loading Tests

    After a test list terminator has been encountered, all the test in the string will now load before stopping for parameters (if monitor Stop/Jump parameter bit 0 - stop to enter parameters was set.)

    a. As monitor initializes each test, they will type out their respective test headings.

    b. After all tests have been initialized, they will stop for parameters if the monitor STJP stop to enter parameters has been set. Otherwise, they will go immediately into execution.

12. Test Parameters

    If monitor STJP has been set, the tests will stop for parameters. The format is as follows:

    A1 = TTN4 (ID) Q1 = STJP
    TT = Test number
    N = The number of stops (pairs of A and Q) including this stop that make up this call.
    1 = Type stop = parameter stop

    STJP = the Stop/Jump parameter for this test only. However, the monitor STJP is logically OR'ed with this Stop/Jump parameter in determining if a test should stop.

    See the respective test documentation for the remaining definitions. On completion of parameter entry, parameters will be typed out on the TTY.

13. Test Execution

    After the parameter stop has been satisfied, the test will go into execution.

    Most all tests can be master cleared and restarted from their initial address (IA = NNNN).

14. Test List Completion

Completion of all selected tests according to the combinations of parameters entered in test sections, test STJP and frequency count, the TTY will type out BUILD TEST LIST. You are now at Step 6.

a. To Change Test STJP Only

Setting the SELECTIVE SKIP switch or pressing manual interrupt will, at some point in time, result in a Monitor Parameter Stop.

A = 0021 Q = STJP

1) Enter test number in upper eight bits of A and the new desired STJP in Q and run. A will then = monitor parameters. Q = equipment address of loader. Ignore this and run again. The new STJP has now been placed for use by that test only.

2) At any stop by the test where A = ID Q = STJP the STJP may be altered for that test (or system) STJP.

3) For purposes of stopping, system Stop/Jump parameter is logically OR'ed with each test STJP when making the decision if to stop for errors, parameters, end of sections and end of tests.

b. Altering The Mask Register

To drop unwanted interrupt levels from the Mask register, there are two avenues open to the user.

1) At end of Quick Look, keeping the SKS switch set will result in a third stop where the user sets the respective bit position in the A register to deselect the respective interrupt line.

2) At any ID stop, select the M register and set or deselect to the desired interrupt configuration. However, once a line has been deselected it cannot be reselected unless monitor is reloaded.

IX. TELETYPE INPUT PACKAGE

The teletype input driver exactly emulates operator panel operations in data entry, register selects, setting the SKIP switch, also doing a master clear, set P, and Run.

A. DATA ENTRY FORMATS

There are three modes of data entry from which the operator can choose for each register entry.

1. Hexadecimal entry

   The data must be prefixed by $ and up to four legal hexadecimal characters, right justified, and zero filled may be entered. Deviation from this will result in an error and restart of the entry.

2. Decimal Entry

   This data is not prefixed and may be up to five legal decimal (0-9) numbers, right justified and zero filled may be entered.

3. Bit Position Selects

   This entry is prefixed by B, then N, N, N where N is a legal decimal number corresponding to respective bit positions separated by commas and in any order.
   Example:
   B, 15, 0, 4, 9, 2, 11 results in $8A15_{16}$

B. DISPLAYED AREA

The data being offered for change is typed out as follows:

$XXXX/$YYYY? Bell CR LF

XXXX = hex contents of the A register
YYYY = hex contents of the Q register

C. DATA ENTRY

The contents of a register is not altered until some data is typed on the TTY. Select one of the three data entry modes and enter the desired data.

After the registers have been typed out (step B), the software pointer is pointing to the A register. If the contents of the register is not to be changed, go to register select or entry termination.

SUPPLEMENT A HAND ENTERED BOOTSTRAPS

A. Bootstraps are included for the following input devices:

1. 1712/13 Teletype Paper Tape Readers
2. 1721/1777 Paper Tape Readers
3. 1731/32 - 601/8 Magnetic Tape (7-Track)
4. 1732 - 609 Magnetic Tapes (9-Track)
5. 1726/405, 1728/430, 1729 Card Readers

B. The equipment address in each bootstrap is indicated by WESD. The operator should enter the equipment address of the applicable loading device.

1712/13 TELETYPE PAPER TAPE

```
XFE0      68FE      TTBOOT    STA*    *-1
XFE1      5803      LDR       RTJ*    FRM
XFE2      011C                SAN     DATA+1
XFE3      1802                JMP*    FRM+1
XFE4      1802      FRM       NUM     $1802
XFE5      6CF9                STA*    (TTBOOT-1)
XFE6      E000                LDQ     =N$WESD WESD=EQUIP ADDR
XFE7      WESD                □       i.e. 0091
XFE8      0A20                ENA     $20
XFE9      03FE                OUT     -1
XFEA      0DFE                INQ     -1
XFEB      CCF3                LDA*    (TTBOOT-1)
XFEC      02FE                INP     -1
XFED      1CF6                JMP     (FRM)
XFEE      58F5      DATA      RTJ*    FRM
XFEF      0FC8                ALS     8
XFF0      58F3                RTJ*    FRM
XFF1      6CED                STA*    (TTBOOT-1)
XFF2      0102                SAZ     ENDBT-*-1
XFF3      D8EB                RAO*    TTBOOT-1
XFF4      18F9                JMP*    DATA
XFF5      1007      ENDBT     JMP-    QL ENTRY
```

The highest core bank for autoload area is indicated by X.

NOTE

Operator instructions on next page.

* 1. Select TTS mode on teletype and load paper tape into teletype paper tape reader.

* 2. Refer to Loading SMM with Hand-Entered Bootstrap and follow that procedure.

* 3. Set switch on teletype paper tape reader to RUN position. Clear BREAK light on teletype if light is illuminated.

4. Quick Look will be loaded and executed (see first flow chart). SMM17 will be loaded.

   * At that time manually stop the reader, select K mode on teletype.

* 5. SMM17 ED. X.X will be typed out. Load teletype paper tape reader with tests to be loaded.

   * Select TTS mode on teletype. Set switch on the teletype paper tape reader to RUN position.

6. PROGRAM PROTECT light will flash while test is being loaded.

   * When loading has completed, select K mode.

7. Test heading will be typed out. Take reader out of RUN position.

8. Test(s) will be executed.

NOTE

Steps under 4, 5, 6, and 7 refer to the 1712 Teletype and are to be ignored if teletype is a 1713.

1721/1777 PAPER TAPE READER

```
XFE0        68FE        PTBOOT      STA*      *-1
XFE1        E000                    LDQ       =N$WESD  WESD=EQUIP ADDR
XFE2        WESD                    □         i.e.  00A1
XFE3        0A20                    ENA       $20       START MOTION
XFE4        03FE                    OUT       -1
XFE5        0DFE                    INQ       -1
XFE6        02FE        LDR         INP       -1
XFE7        0112                    SAN       DATA+1
XFE8        18FD                    JMP*      LDR
XFE9        02FE        DATA        INP       -1
XFEA        0FC8                    ALS       8
XFEB        02FE                    INP       -1
XFEC        6CF2                    STA*      (PTBOOT-1)
XFED        0102                    SAZ       ENDBT-*-1
XFEE        D8F0                    RAO*      PTBOOT
XFEF        18F9                    JMP*      DATA
SFF0        1007        ENDBT       JMP-      QL ENTRY
```

The highest core bank for autoload area is indicated by X.

1731/32/32-2 - 601/8 MAGNETIC TAPE (7-TRACK)

```
XFE0       68FE       MTBOOT     STA*     *-1
XFE1       E000                  LDQ      =N$WESD  WESD=EQUIP ADDR
XFE2       WESD       EQUIP      □        i.e. 1382
XFE3       C000                  LDA      =N$405
XFE4       0405                  □
XFE5       03FE                  OUT      -1    Select Unit 0 and Binary
XFE6       09FB                  INA      -4
XFE7       0DFE                  INQ      -1
XFE8       03FE                  OUT      -1    Rewind
XFE9       0F42                  ARS      2
XFEA       03FE                  OUT      -1    Motion
XFEB       0DFE                  INQ      -1
XFEC       0A00                  ENA      0
XFED       1807                  JMP*     MT2
XFEE       02FE       MT1        INP      -1
XFEF       0FC6                  ALS      6
XFF0       BCEE                  EOR*     (MTBOOT-1)
XFF1       0109                  SAZ      ENDBT-*-1
XFF2       7CEC                  SPA*     (MTBOOT-1)
XFF3       D8EB                  RAO*     MTBOOT-1
XFF4       02FE       MT2        INP      -1
XFF5       7CE9                  SPA*     (MTBOOT-1)
XFF6       02FE                  INP      -1
XFF7       0FCA                  ALS      10
XFF8       BCE6                  EOR*     (MTBOOT-1)
XFF9       7CE5                  SPA*     (MTBOOT-1)
XFFA       18F3                  JMP*     MT1
XFFB       1007       ENDBT      JMP-     QL ENTRY
```

The highest core bank for autoload area is indicated by X.

NOTE

Load the SMM17 Magnetic Tape on unit 0 and ready the unit.

1732/1732-2 - 609 MAGNETIC TAPE (9-TRACK)

```
XFE0     68FE     MTBOOT   STA*   *-1
XFE1     E000              LDQ    =N$WESD WESD=EQUIP ADDR
XFE2     WESD     EQUIP    □      i.e. 1382
XFE3     C000              LDA    =N$44C
XFE4     044C              □
XFE5     03FE              OUT    -1     SELECT UNIT 0
XFE6     09B3              INA    -$400-$44C
XFE7     0DFE              INQ    -1
XFE8     03FE              OUT    -1     REWIND
XFE9     0F42              ARS    2
XFEA     03FE              OUT    -1     MOTION
XFEB     0DFE              INQ    -1
XFEC     02FE     MT1      INP    -1
XFED     6CF1              STA*   (MTBOOT-1)
XFEE     0102              SAZ    ENDBT-*-1
XFEF     D8EF              RAO*   MTBOOT-1
XFF0     18FB              JMP*   MT1
XFF1     1007     ENDBT    JMP-   QL ENTRY
```

The highest core bank for autoload area is indicated by X.

1726/405, 1728/430, 1729 CARD READER

```
XFE0        0AFE        CRDBOOT    ENA      -1
XFE1        68FD                   STA*     *-2
XFE2        E000                   LDQ      =N$WESD WESD=EQUIP ADDR
XFE3        WESD        EQP1       □        i.e. 0621
XFE4        C812        J430       LDA*     FCN
XFE5        03FE                   OUT      -1
XFE6        0DFE        J405       INQ      -1
XFE7        0206                   INP      EXT
XFE8        0FC8                   ALS      8
XFE9        D8F5                   RAO*     CRDBOOT-1
XFEA        6CF4                   STA*     (CRDBOOT-1)
XFEB        02FE                   INP      -1
XFEC        BCF2                   EOR*     (CRDBOOT-1)
XFED        6CF1        INT        STA*     (CRDBOOT-1)
XFEE        E8F4        EXT        LDQ*     EQP1
XFEF        02FE                   INP      -1
XFF0        0FCB                   ALS      11
XFF1        0131                   SAM      EOP-*-1
XFF2        18F3                   JMP*     J405
XFF3        CCEB        EOP        LDA*     (CRDBOOT-1)
XFF4        0102                   SAZ      ENDBT-*-1
XFF5        18EE                   JMP*     J430
XFF6        0081        FCN        □
XFF7        1007        ENDBT      JMP-     QL ENTRY
```

NOTE

Tests selected to be run under SMM17 should be placed in the input tray of the card reader in the same order as selected. If not in the same order, the loader will skip cards until it finds the test selected or detects a Hopper Empty condition.

The highest core bank for autoload area is indicated by X.

When using 405 card reader locations XFF5 should be 18F0, and location XFF6 should be 0401.

SUPPLEMENT B SMM17 LIBRARY FORMAT

Library program data is formatted similar to that of MSOS 1700 Assembler output, RBD (relocatable binary data), except for the first record. This record (absolute binary Quick Look) consists of the actual core image of QL and the attached intermediate loader with the Hex-words broken down as is convenient for a simple bootstrap loading program for a particular device. For example, card, paper, and 9-track mag. tape use a 2x8-bit breakdown while 7-track mag. tape bit sequence is lower 6, upper 6, then middle 4 for each Hex-word.

RBD is blocked data with flags embedded in the block to signal biasing or relocation by SMM's RBD loader according to the load FWA (test IA). Thus the program under load becomes relocatable by modifying the necessary data words and addresses in the program to reflect the program's core position. Block length is one card image or less. Block type is identified by its first word:

NAMBLK (Program name block)	= $2050
RBDBLK (Program data block)	= $4050
BZSBLK (Zero data block)	= $6050
XFRBLK (Transfer, end of program)	= $C050
OVRBLK (Overlay segment marker)	= $E050, unique to SMM17
HEXBLK (Program Modification data)	= $2A48

(Other block types are considered illegal by the loader as an SMM program cannot specify entry points or externals, $8050, $A050 respectively. See MSOS Reference Manual.)

Block construction is as follows:

NAMBLK (derived from NAM card of program, and assembly information)

Word #1 = Block ID = $2050

2 = Zero for SMM, common storage word count for MSOS

3 = Data area (set=BZS area by pseudo-op to aid SMM DP loader)

4 = Program length (LWA=1 of last program data word assembled)

5 = Characters 1, 2

6 = Characters 3, 4 } Program name in ASCII code

7 = Characters 5, 6

Words 11-29 = Program revision, copyright dates (NAM card columns 25-62)

Word 30 = Blanks or 01 (ASCII) where 1=flag loader to ignore relocation on load (NAM card columns 63-64)

RBDBLK

Word 1 = Block ID=$4050

2 = R0, R1, R2, R3* = Relocation Flags for W0-W3

3 = W0 = Starting address of program data in this block

4 = W1
5 = W2
6 = W3

Program data words

Word 52 = R40, R41, R42, R43 = Relocation Flags for W40-W43

53 = W40
54 = W41
55 = W42
56 = W43

Program data words

BZSBLK

Word 1 = Block ID = $6050

This block is ignored by SMM's RBD loader. The program load area is zeroed according to length in NAMBLK before reading data blocks (other block words are unimportant to SMM).

XFRBLK

Word 1 = ID = $C050

Words 2 through 4 are unimportant to SMM.

OVRBLK

Word 1 = ID = $E050

2 = 0VXX, where V=overlay segment number, XX=test ID number

3 = Overlay first word address

4 = Unimportant

*Only RX values of 0/1, = no bias/bias, are used in SMM. The upper bit of the Hex-digit is set to flag last program data word of block (for short blocks).

HEXBLK

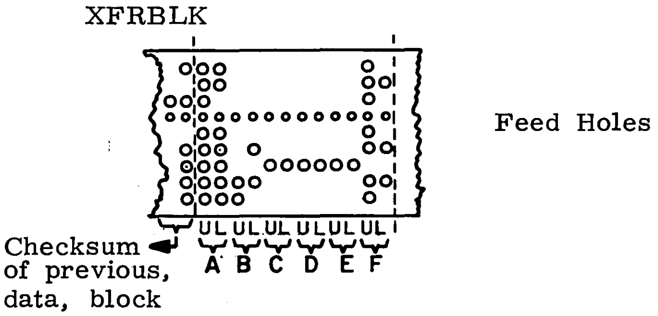
Same as RBDBLK except for word 1.

Word 1 = ID = $2A48 (* H, ASCII)

Device Format Variance

Paper tape RBD records include a prefix word and a suffix word. The prefix word is the complemented Hex-word length of the block, not including the prefix and suffix words. The suffix word is the entire block checksum word, including prefix and suffix.

Paper tape example:



A = Complement count, B-E

B = Block ID, $C050

C = ASCII blanks = $2020

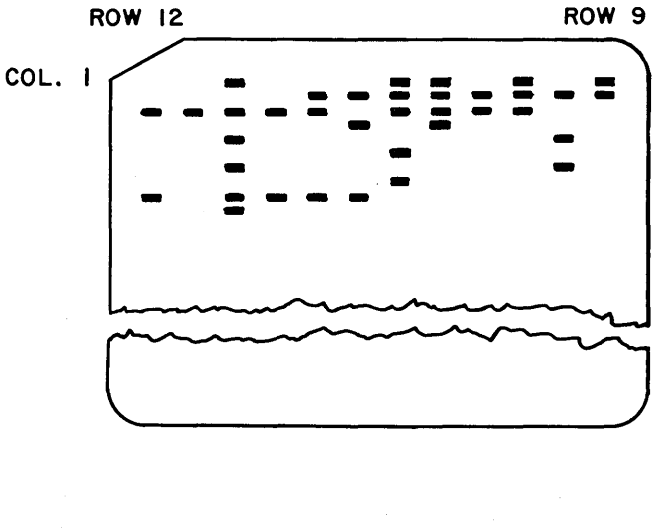
D = ASCII blanks

E = ASCII blanks

F = Complement sum, A-E (checksum)

Card image reads from row 12 to row 9, column 1 through column 80, to give Hex-words, upper through lower bits respectively (turn card upside down, column 1 to top). The first Hex-word, however, is broken up into an 8-bit card sequence number (upper bits), a 4-bit Hex-digit signifying binary punched card (7, 9 punched), and a last digit used by MSOS but not SMM. The second Hex-word is the complemented RBD block length and does not include these first two words, nor the last, card checksum word. The checksum word includes the entire card.

Card example:



1st two Hex-digits = Card Sequence = $23

3rd Hex-digit = Binary (7/9) punch

4th Hex-digit = Zero

5th through 8th Hex-digits = Complement Block word connt = $FFEB

9th through 12th Hex-digits = Block ID = $C050

13th through 16th Hex-digits = ASCII blanks = $2020

17th through 20th Hex-digits = ASCII blanks

21st through 24th Hex-digits = ASCII blanks

25th through 28th Hex-digits = Complement sum of all previous Hex-words (checksum)

SUPPLEMENT C QUICK LOOK COMMAND TEST

Quick Look is an abbreviated command test which checks out the instruction set of the 17X4 before SMM17 is loaded. Quick Look is automatically loaded and executed each time the monitor is loaded. Quick Look cannot be called as an individual test.

OPERATION

Quick Look executes as follows:

1. Control is given to Quick Look at either location $0 or $7.
2. A checksum is performed as part of the initialization. Whenever the checksum is not equal to zero a stop will occur with P register equal to $001F.
3. Quick Look can be executed even if a checksum error exists. Set RUN/STEP switch to RUN in order to ignore the error. (False error stops may occur when executing Quick Look since a checksum error denotes either that Quick Look is not residing in core correctly or that the instructions listing the checksum fails.)
4. Whenever an error stop occurs in Quick Look, check the address displayed in P register and refer to QUICK LOOK ERRORS.
5. Running after an error stop has occurred will cause the failing portion of Quick Look to be repeated.
6. Quick Look is programmed to allow restart by a Master Clear and Run. All instructions are executed and checked in Quick Look except the following:

   NOP
   EIN
   EXI
   SPB
   CPB

QUICK LOOK ERRORS

All error stops in Quick Look are unconditional stops.

The following is a list of addresses (ADDR) that appear in P register whenever instructions fail in Quick Look. The instruction associated with each address is listed next to it. A listing should be consulted to further determine the cause of the error.

ADDR	INST	ADDR	INST	
001F	CHSM*	00C4	SPA	
0024	ENA or SAZ	00CA	SPA	
0028	ENQ or SQZ	00D4	SPA	
0026	SAN	00DB	SPA	
002F	SQN	00E2	TRQ A	
0035	SAZ or ENA	00EA	LDQ	
0038	SQZ	00F3	STQ	
003B	SAN	00FE	RAO	
003F	SQN	010A	ADQ	
0045	SAP	0118	SET M or TRM A	
0049	SQP	011E	SET A	
004D	SAM	0122	SET Q	
0050	SQM	0129	TRQ A	
0057	SAP	0131	TRQ M	
005A	SQP	0138	TRM Q	
005D	SAM	013E	TRA Q	
0061	SQM	0146	TRA M	
0067	EOR	014A	SWS or SWN	
006F	EOR	014E	SWN or SWS	
0076	LDA	0153	SOV	
007E	AND	0156	SNO	
0086	AND	015C	SOV	
008C	SUB	0163	SNO	
0094	SUB	0168	SPE or SNP	
009E	ADD	016C	SPE or SNP	
00A9	ADD	0172	INP or INT REJ	Q = EQADR
00B3	STA	0179	OUT or INT REJ	
00BD	STA	017C	EXT or INT REJ	

SUPPLEMENT D

SMM1700 PROGRAMMING SPECIFICATIONS

I. INTERFACING WITH SMM

A. TEST STRUCTURE

1. The NAM Card Format:

   a. Columns 2-4 = NAM

   b. Columns 12-14 = (Test Mnemonic) i.e. COM

   c. Column 15 = A (for run alone) or Zero (0), (for multiplexed with other test)

   d. Columns 16-17 = (Test Number) i.e. 01

   e. Columns 25-26 = Month revised, i.e. 06 = June

   f. Columns 27-28 = Date revised, i.e. 15 = 15th

   g. Column 29 = Last digit of year revised, i.e. 2 = 1972

   h. Columns 31-62 = Copyright data as follows:
      COPYRIGHT CONTROL DATA CORP 1970

   i. Column 64 = Blank or Zero (0); relocation is to be done by the monitor. (Bias value NOT calculated by the test).

Example:

```
NAM       COM001    06152 COPYRIGHT CONTROL DATA CORP 1970
```

2. a. The test must not contain an ORG statement to an absolute address.

   b. An ORG * statement (not ORG*) immediately preceding the identification (ID) word of the test's parameter data block is required to facilitate prestored parameter modification by the editing routine. (I. A. 4c)

   c. An ORG statement resulting in program address reversal is used at the beginning of an overlay routine and flags the editing routine to insert an overlay marker block in the edited library. The overlay marker block both terminates primary load and identifies the overlay on a subsequent overlay call. Overlay calls are of the form RTJ-

(OVRLAY) with A REG = Overlay Number. The routine returns to the user at P + 1 for a successful load, or P + 2 on an unsuccessful load (terminal load error encountered). The LWA of the test must be set to a value sufficient to accommodate the longest overlay to prevent overlaying the next test.

The last program address of the final overlay must be equal to the LWA of the entire test. This is used by the assembler to determine program length for RBD NAMBLK information.

3. All references to the monitor must be made through the use of tags that are equated to absolute low core locations. This block of EQU cards must follow the NAM card. The equate deck structure is indicated in I.I (equates).

4. The first locations of the test must contain the following:

   a. INITIAL ADDRESS (IA) = A one-word jump (JMP*) instruction to the Master Clear restart address. The tag for this jump instruction should be the same 6 characters used in columns 12-17 of the NAM card.

   b. IA+1 through IA+3 = An ALF statement of the 6 characters contained in columns 12-17 of the NAM card, i.e., ALF 3, COM001.

   c. IA+4 = An ADC statement of the Parameter Entry ID word, i.e., ADC IDWRD.

   d. IA+5 = An ADC statement of the entry to the initialization routine. The monitor always transfers control to the test at IA+5, therefore the test must ensure that its next return address is stored at IA+5 prior to returning control to the monitor.

   e. IA+6 = Equipment address for Director Status 1. This location may be prestored with the equipment address; if so, it may be changed or entered during loading of the test.

EXAMPLE

```
COM001    JMP*     PARAM         PARAMETER ENTRY
          ALF      3, COM001
PARADR    ADC      TAG           TAG=PARAM ID WORD
RETURN    ADC      INITIAL       INITIALIZATION
EQUIP     NUM      0
```

5. After IA+6 the test will have subroutines and sections. The following is a list of possible subroutines.

   a. Repeat Condition

   b. Repeat Section

   c. Repeat Test

   d. Section Selection

   e. Parameter Routine

   f. End of Test

   g. Error Reporting

   h. Return Control to Monitor

   i. Status

   j. Function Selection

   k. Data Output

   l. Data Input

   m. Interrupt Processor

   n. Counters

6. The last portion of the test will be initialization followed by the END card.

B. INITIALIZATION

1. This portion of the test can be used for a data storage area by the test.

2. This routine is entered via a transfer of control from the monitor to the address (ADC) at IA+5.

3. The following are functions this routine performs:

   a. Convert IA and frequency count to ASCII, and store in title message.

   EXAMPLE

```
LDA       =XCOM001
RTJ-      (HEXASC)
STQ       Location in title for upper 2 characters
STA       Location in title for lower 2 characters
LDQ-      TSACTV
```

```
LDA-    TSFREQ-1,Q
RTJ-    (HEXASC)
STA     Location in title for frequency count
```

b. Change location IA+5 to the address (parameter entry routine) where control is to be returned to the test by monitor. (Run-alone test does not return control to monitor until completion of execution.)

c. Type out program name, initial address, and frequency count.

d. Execute RTJ- (CONTROL) to return control to the monitor.

e. After control is returned to the test the parameter entry routine is executed.

f. Interrupt line is requested, i.e., RTJ- (REQINT)

g. Reset return address (IA+5) to the address of test section control.

h. Execute RTJ- (CONTROL) to exit initialization except for run-alone tests.

C. CONTROL

1. Loading

a. After loading all tests called or a run-alone test is loaded, the tests are then sequentially given control at location IA+5. After initializing itself the test returns control via RTJ- (CONTROL). This process is continued until all tests are initialized.

2. Execution

a. Control is passed to a test indirect through IA+5 during execution.

b. After the test receives control from the monitor, the test must decide whether control can be accepted, that is, is the unit under test capable of continuing testing at this time. If control cannot be accepted by the test, control is returned to the monitor.

c. The test returns control to the monitor via an RTJ- (CONTROL). Before passing control to the monitor, the test must store the address where the monitor should return control at test location IA+5.

d. The test should return control to the monitor as often as possible, but not at times that could cause conflicts due to interaction of tests, creating false errors. A run-alone test does not return control to the monitor until it has completed execution. The test must not execute an RTJ- (CONTROL) while in its interrupt processor.

e. When test has completed execution, an RTJ- (EXIT) is executed, after storing restarting return address in IA+5. If the frequency number was greater than 1, control is returned at the return address IA+5.

f. Restarts may be made by Master Clear, set P equal to IA and RUN.

D. STOP AND JUMP

The test must use the conditional stop and jump routines provided by the monitor. Conditional stops and jumps are selected by the operator setting appropriate bits in the Stop/Jump (S/J) parameter word. Each test is provided with its own (SJ) parameter word. This word is displayed in Q whenever a conditional stop is encountered. The test CANNOT ENTER THE STOP AND JUMP ROUTINE WHILE IN INTERRUPT STATE.

1. Conditional Stops in Stop/Jump Parameter

a. SMM17 provides four programmed stops specified by bits 0 through 4 of Stop/Jump parameter.

1) Stop 1 (Bit 0) - Stop to enter parameter.

2) Stop 2 (Bit 1) - Stop at End of Test section.

3) Stop 4 (Bit 2) - Stop at End of Test.

4) Stop 8 (Bit 3) - Stop on error.

b. Test does a RTJ- (STOP) with the Identification (ID) word and display information in consecutive locations starting at return address plus one. If the Stop switch is set and the operator selected the stop by setting the appropriate bit in the Stop/Jump parameter word, a stop will occur in the monitor. The format in which data is displayed must conform to the specifications of the individual stop. The first stop will always display the Identification Word in A and the Stop/Jump parameter word in Q.

1) STOP 1 - Stop to enter parameters

Bit 0 of Stop/Jump parameter and ID words must be set for Stop 1 to be executed. Upon stopping, parameters are entered into A and Q in consecutive order until all parameters designated by the number of stops in the ID word have been entered. The display of data begins after the first stop. The data that is presently in the parameter area will be displayed. A set of commonly used parameters (normally for executing without operator intervention) must be prestored in the parameter area, or stored during editing.
Test Bias = The initial address of the test.

EXAMPLE

```
LDA      BIAS       The Initial Address
RTJ-     (STOP)     To stop routine
JMP*     LABEL
ORG      *
NUM      $XXYZ      ID word (SMM adds $10_{16}$)
                    XX = test number
                     Y = number of stops (excluding ID and S/J
                         stop)
                     Z = type of stop (Bit 0 set for Parameter stop)
PARAM       NUM     $AAAA     Test parameter area (maximum
            NUM     $BBBB     number of parameters = $28_{10}$)
LABEL       Continue
If ID = $0C11
SMM displays:  A = 0C21
               Q = Stop/Jump
               A = AAAA
               Q = BBBB
```

2) STOP 2 - Stop at End of Section

Bit 1 of Stop/Jump parameter and ID word must be set for Stop 2 to be executed.

EXAMPLE

```
LDA         BIAS        Load A with test bias to stop routine
RTJ-        (STOP)      ID word (SMM adds $20_{16}$)
JMP*        LABEL
NUM         $XXYZ       XX = test number
                         Y = number of stops (excluding ID and S/J
                             stop)
                         Z = type of stop (Bit 1 set for End of Section
                             stop)
NUM         $SS00       Section number
LABEL  Continue
If ID = $1502 and section number = $06 SMM displays:
           A = 1522
           Q = Stop/Jump
           A = 0600
           Q = Return address
```

3) STOP 4 - Stop at End of Test

Bit 2 of Stop/Jump parameter and ID words must be set for Stop 4 to be executed.

EXAMPLE

```
RAO*     PASS       Update pass count
LDA      BIAS       Load A with test bias to stop routine
RTJ-     (STOP)
JMP*     LABEL
NUM      $XXYZ      ID word (SMM adds $20_{16}$)
                    XX = test number
                     Y = number of stops (excluding ID and
                         S/J stop)
                     Z = type of stop (Bit 2 set for End of
                         Test stop)
PASS     NUM        $0005     Pass count
LABEL    Continue
```

If ID = $0C04 SMM displays:

A = 0C24
Q = Stop/Jump
A = 0005
Q = Return address

4) STOP 8 - Stop on error

Bit 3 of Stop/Jump parameter and ID words must be set. The display of data begins at the third stop.

EXAMPLE #1

```
LDA      BIAS       Load A with test bias to stop routine
RTJ-     (STOP)
JMP*     LABEL
NUM      $XXYZ      ID word (SMM adds $20_{16}$)
                    XX = test number
                     Y = number of stops
                     Z = type of stop (Bit 3 set for error stop)
NUM      $SSEE      Section/Error numbers
                    SS = Section error occurred in
                    EE = Error number
NUM      $AAAA      Test Data 1
NUM      $BBBB      Test Data 2
LABEL    Continue
```

If ID = $4F18, section number = 07, and error number = $21 SMM displays:

```
A = 4F38
Q = Stop/Jump
A = 0721
Q = Return Address
A = AAAA
Q = BBBB
```

c. The monitor determines the return address. A register is loaded with the address bias value before entering the stop routine. The monitor determines whether the return address displayed is to be the listing address or memory address (bit 9 of S/J parameter word). The return address is the location within the test from which the test stop routine was called.

d. When a common error reporting routine is used a location is provided in the stop routine for the return address. The return address is established by an RTJ entry to the common error stop routine which is determined and stored by the test, in the location provided. Before storing, the test must check bit 9 of the Stop/Jump parameter and determine if it is to be stored with or without bias value. The A register is cleared when the call to RTJ- (STOP) is made.

EXAMPLE 2

```
ERROUT    NUM     0            Common error stop routine entry
          LDA*    ERROUT       Get return address (biased)
          LDQ-    STJP
          QLS     6            Check bit 9
          SQM     STRTA-*-1    Bit 9 set, store with bias
          SUB     BIAS         Bit 9 not, set subtract bias
STRTA     STA     RTNADR       Store return address
          CLR     A            Clear A
          RTJ-    (STOP)
          JMP*    LABEL
          NUM     $XXYZ        ID word (SMM adds 20_16)
                               XX = test number
                                Y = number of stops
                                Z = type of stop (bit 3 set for
                                    error stop)
```

```
         NUM      $SSEE        Section/Error numbers
                               SS = section error occurred in
                               EE = error number
RTNADR   NUM      $RRRR        Return address
         NUM      $AAAA        Test data 1
         NUM      $BBBB        Test data 2
         LABEL    Continue
```

2. Conditional Jumps in Stop/Jump Parameter

   a. SMM provides 12 programmed jumps specified by bits 4 through 15 of Stop/Jump parameter.

      1) Jump 0 (Bit 4) - Repeat conditions

      2) Jump 1 (Bit 5) - Repeat section

      3) Jump 2 (Bit 6) - Repeat test

      4) Jump 3 (Bit 7) - Not used

      5) Jump 4 (Bit 8) - Omit typeout

      6) Jump 5 (Bit 9) - Bias "Return Address " display

      7) Jump 6 (Bit 10) - Re-enter parameters

      8) Jump 7 (Bit 11)
      9) Jump 8 (Bit 12)
      10) Jump 9 (Bit 13)
      11) Jump 10 (Bit 14)
      12) Jump 11 (Bit 15)

      } Optional jumps determined by individual test

b. Test does an RTJ- (JUMP) with the correct jump mask in A. If the appropriate bit in A register matches the Stop/Jump parameter, control is returned to P (P = return jump address in the test). If a comparison is not found, control is returned to P+1.

EXAMPLE

```
ENA       $10       Repeat conditions bit set?
RTJ-      (JUMP)
JMP*      LABEL     Yes - Set up and repeat condition just executed
Continue            No - Continue on to next condition
```

c. Care must be taken to check conditions of Stop/Jump word in logical order, that is, Stop at End of Section (Bit 1) should be checked before Repeat Section (Bit 5) is checked.

3. Stop and Skip Switches

Selective stop and skip switches are used to bring about certain conditions in the operation of the monitor and tests. The tests cannot use these switches for any other purpose than defined in these specifications.

a. Selective Stop Switch should always be set.

b. Selective Skip Switch is set by the operator to display the monitor parameters: Stop/Jump Parameter, SMM Parameter and Equipment Address for the resident loader.

E. INTERRUPTS

Test must request permission to select an interrupt line from SMM17. This request cannot be done while in interrupt state. SMM17 gives the test permission to use a particular interrupt line after checking for a previous selection of the same line. SMM17 DOES NOT SELECT THE INTERRUPT.

1. Test requests permission to select an interrupt line by executing the following procedure:

a. Load appropriate interrupt line parameter into the A register, i.e., line 6 equals bit 6 set. This interrupt line parameter is from the parameter area.

b. Load the equipment address for status in Q register for the unit to be tested. (Contents of test initial address +6).

c. Test does a RTJ- (REQINT).

d. The location following the return jump instruction must contain the address of the test's interrupt processor.

EXAMPLE

```
LDA          INTLIN        Interrupt line to be requested
LDQ          EQUIP         Equipment address
RTJ-         (REQINT)      To request interrupt
ADC          INTPRO        Address of interrupt processor
Continue
```

e. Only one test can request a given interrupt line. The test must be able to handle the common interrupt line or multiple interrupt lines if the unit under test has this feature. The test interrupt processor must determine which interrupt occurred in the situation where more than one unique equipment interrupt is on the same line.

f. The functions described in a through c above must be performed when a Master Clear restart is executed.

2. Interrupt processing is accomplished in the following manner:

a. SMM saves the contents of A, Q, I and M registers.

1) Interrupt on lines two to fifteen - SMM stores the interrupt processor address for the line that interrupted in the I register.

2) Interrupt on line 1 - SMM checks for an interrupt condition for each unit in the slow speed package. If a station is found that interrupted, SMM stores the interrupt processor address for that unit in the I register and passes interrupt control to it. (See step c.) If no slow-speed unit interrupted, the interrupt processor address for that line is stored in the I register and interrupt control is passed to it (See step c.)

3) Interrupt is on line 0 - and there has been a request for that line, SMM passes interrupt control to it. (See step c.) If no request has been made, control is passed to the SMM17 parity and protect fault routine for error output.

b. SMM clears the interrupt mask bit corresponding to the interrupt, enables interrupts and does a return jump indirect to the contents of the I register with Q equal to the exit interrupt value for that line.

c. The test processes the interrupt and passes interrupt control back to SMM via the return address with the exit interrupt value in Q.

d. SMM restores A, Q, I and M registers and exits interrupt state.

F. MESSAGES

Test communications to the operator must use the monitor TYPEOUT routine for outputting messages. Typical messages would include program name and initial address and operator intervention messages such as Clear Protect Switch. This routine should not be used while in interrupt state.

Message TYPEOUT routine types out ASCII code packed two characters per word on teletype and is executed by using the following procedure:

1. Reset return address (IA+5) to loop address.
2. The BUSY switch is checked by the TYPEOUT routine, therefore it is not necessary for the test to perform this task.
3. Monitor will set BUSY switch - Bit 0 when the test enters the TYPEOUT routine.
4. Load A with the first word address of data to be typed out.
5. Load Q with the number of words to be typed out.
6. Do an RTJ- (TYPEOUT). Monitor will clear BUSY switch Bit 0 to inform other tests when they may use TYPEOUT routine.

EXAMPLE

```
            LDA        =XLABEL0
            STA        RETURN         (IA+5)
LABEL0      ENA        1
            AND-       INFORM         Check BUSY switch
            SAZ        LABEL1-*-1     subroutine to return
            RTJ-       (CONTROL)      control to monitor
LABEL1      LDA        =XDATA         Data address (biased)
            ENQ        10             Number of words
            RTJ        (TYPEOUT)      TYPEOUT routine
            Continue
```

G. SUBROUTINES AVAILABLE TO TEST

The following are subroutines available to the test. These subroutines cannot be used while in interrupt state.

1. Convert hexadecimal numbers to ASCII codes.
   a. Load A with numbers to be converted.
   b. Do a RTJ- (HEXASC).
   c. When control is returned to subprogram, Q equals the upper two characters and A equals the lower two characters.
2. Generate Random Numbers
   a. Load Q with the number of words to be generated.
   b. Load A with the first word address of area where numbers are to be stored.
   c. Do a RTJ- (GENRAN).

H. INFORM WORD

The information word is a monitor low core location which contains the Busy switches, M register size and core size. The information word is divided as follows:

1. Bit 0 Busy switch for teletype
2. Bit 1 Busy switch for paper tape reader
3. Bit 2 Busy switch for paper tape punch
4. Bit 3 Busy switch for card reader
5. Bit 4 Busy switch for 1716, 06 No. 1
6. Bit 5 Busy switch for 1716, 06 No. 2
7. Bit 6 Busy switch for 1716, 06 No. 3
8. Bit 7
9. Bit 8
10. Bit 9
11. Bit 10
12. Bit 11 Mask Size
    a. 0 = 4 Bit mask
    b. 1 = 16 Bit mask

13. Bit 12 to 15 - Core size

a. 0000 = 4K

b. 0001 = 8K

c. 0010 = 12K

d. 0011 = 16K

e. 0100 = 20K

f. 0101 = 24K

g. 0110 = 28K

h. 0111 = 32K

I. EQUATES

List of monitor location equates for test use are:

```
EQU     CONTROL (1)             RETURN SMM CONTROL ADDRESS
EQU     STOPX(CONTROL+1)        ALL STOPS AND ERROR TYPEOUTS
EQU     EXIT(STOPX+1)           END OF TEST-PASS EXIT
EQU     REQINT(EXIT+1)          REQUEST INTERRUPT ADDRESS
EQU     FCLRINT(REQINT+1)       FAKE CLEAR INTERRUPT ROUTINE
EQU     JUMP(REQINT+2)          CHECK SKIP SWITCH FOR PARAM ENTRY
EQU     GENRAN(JUMP+1)          RANDOM NUMBER GENERATOR
EQU     TYPEOUT(GENRAN+1)       MESSAGE TYPEOUT ROUTINE
EQU     TTYBZY(TYPEOUT+1)       ROUTINE TO WAIT TTY NOT BUSY
EQU     HEXASC(TYPEOUT+2)       HEX TO ASCII CONVERSION
EQU     OVRLAY(HEXASC+1)        OVERLAY CALL ROUTINE
EQU     RELPOS(OVRLAY+1)        A/Q RELATIVE POSITION TEST
EQU     MAINL(RELPOS+1)         MANUAL INTERRUPT ROUTINE
EQU     SFTMASK(MAINL+53)       M REGISTER RESET VALUE
EQU     STJP(SETMASK+1)         STOP/JUMP PARAMETER WORD
EQU     LASTVALU(STJP+1)        MONITOR AREA LWA+1
EQU     LASTAD(STJP+2)          AVAILABLE LOAD CORE FWA
EQU     LDLCORE(LASTAD+2)       BANK0 LAST ADDRESS
EQU     LDL1COR(LDLCORE+1)      BANK1 LAST ADDR
EQU     INFORM(LDL1COP+1)       BK0 SIZE, MASK SIZE, BUSY SWITCH
EQU     SMMCNT(INFORM+13)       SMM PARAMETER WORD
EQU     BIT00(SMMCNT+21)        XXXXXXXX CONSTANT TABLE XXXXXXXX
EQU     BIT0(BIT00)
EQU     BIT1(BIT0+1)
EQU     BIT2(BIT1+1)
EQU     BIT3(BIT2+1)
EQU     BIT4(BIT3+1)
EQU     BIT5(BIT4+1)
EQU     BIT6(BIT5+1)
EQU     BIT7(BIT6+1)
EQU     BIT8(BIT7+1)
EQU     BIT9(BIT8+1)
EQU     BIT10(BIT9+1)
EQU     BIT11(BIT10+1)
EQU     BIT12(BIT11+1)
EQU     BIT13(BIT12+1)
EQU     BIT14(BIT13+1)
EQU     BIT15(BIT14+1)
EQU     H0000(BIT15+1)
EQU     HFFFF(H0000+1)
EQU     H000F(HFFFF+1)
EQU     H00F0(H000F+1)
EQU     H0F00(H00F0+1)
EQU     HF000(H0F00+1)
EQU     H00FF(HF000+1)
EQU     HFF00(H00FF+1)
EQU     HFFF0(HFF00+1)
EQU     H0FFF(HFFF0+1)
EQU     HFF0F(H0FFF+1)
EQU     HF0FF(HFF0F+1)
EQU     H7FFF(HF0FF+1)
EQU     H7F00(H7FFF+1)
EQU     H0780(H7F00+1)
EQU     H007F(H0780+1)
EQU     H2020(H007F+1)
EQU     H2020(H007F+1)
EQU     TSACTV(H2020+6)          TABLE INDEX TO TEST IN CONTROL
EQU     TSFREQ(TSACTV+1)         TEST, FREQUENCY TABLE
```

The test must not use any low-core locations for storage use. Low-core is restricted to monitor use. This does not include location $00FF, i.e., I register.

II. DESIGN SPECIFICATION

A. SECTION STRUCTURING

1. Sections should be structured so as to allow any section to be executed by itself or following any other section. Section must be repeatable and, therefore, conditions must be initialized before each pass. The routine for selecting

sections should allow for the addition of more sections in the future. Sections should call on common subroutines as often as possible. Control must be returned to the monitor between sections.

2. One section should be singled out as an interrupt test although other sections may also use interrupts.

3. At least one section should check buffered data transfers in case the equipment is on a 1706 or 1716.

4. Conditions must be repeatable whether an error occurred or not. Each condition should be a complete entity in itself. All initial conditions must be included in the loop. Channel and non-channel considerations must be taken into account. Control must be returned to the monitor between conditions.

5. The sections must be written such that a Master Clear on the computer can be done at any time. After the Master Clear and restarting the test at IA, the test must still execute correctly.

B. STATUS

1. A full status word compare must always be used. Status should be taken and two compares of the bits made, check for those bits that are expected to change and then a check for those bits that are not expected to change.

2. Status operation should be done in a subroutine rather than in line code.

3. Both the channel and equipment status should be checked when applicable. The channel status must be taken before the equipment status. (If an equipment status attempt is made when the channel is Busy, a reject will occur.)

4. When test is in Wait mode because of positioning, I/O, timing, etc., the program should be monitoring the full status word, not only the single condition bit. Abnormal and other erroneous status conditions could be missed if this check is omitted.

C. HANGING

1. The test must ensure that the I/O channel is protected from hanging up. Some form of time out along with status checking (full word) should be used.

2. To prevent hanging the system, the test must not hang on an INPUT or OUTPUT instruction i.e., INP -1. All rejects should use some form of time out or retry counts along with status checking.

D. RETRY CONSIDERATIONS

The test should have two selectable modes of operations where applicable.

1. Report all errors and allow normal repeat condition.
2. When detecting a data or parity error, repeat the Read or Write operation N times before reporting error. If retry corrected the error, update the number of retries required.

The test can repeat an erroring operation N times before reporting the malfunction. This is usually done only on reading and writing of data. The number of times the retry procedure was used must be updated and reported when the test is completed.

E. ERROR REPORTING

All available pertinent information should be reported in the error display. This information should be gathered at the closest possible time to the moment of the erroring condition. Examples of error information to report follow:

1. Channel conditions - All status
2. Equipment conditions - All status
3. Data written and data read
4. Timing
5. Retries
6. Address at which error was detected
7. Contents of pertinent registers
8. Interrupts selected and expected
9. Critical locations in test

F. TIMING CONSIDERATIONS

1. Running Time
   a. Total time to execute the entire test with the prestored parameters should be available in the test writeup.
   b. Total time to execute each section of the test should be available in the test writeup.
2. Timing of Operations
   a. Timing must take into consideration the differences in configurations, that is, with a channel or without a channel.
   b. For accurate timing in critical portions of the test, it may be necessary to run this portion by itself, that is, not with other tests.
   c. Timing must take into consideration the differences between the 1700 processors (1704/1714, 1774, and 1784) regarding the differences in in-

struction execution time. A test must execute properly on either of the processors unless it is designated to be restricted to a particular processor. Bits 2 and 3 of the SMM parameter denote the various memory speeds of the processors as follows:

Conditions of Bits 3	2	Processors	Memory Speed
0	0	1704/1714	1.1 microseconds
0	1	1774	1.5 microseconds
1	0	1784	900 nanoseconds
1	1	1784	600 nanoseconds

G. TAG REFERENCES

Tags must always be used by the test in place of relative numeric values when referencing test and monitor locations. Involved are the SKIP, I/O, and all Memory Reference instructions. If tags are not used, adding and/or deleting op-codes from the source could necessitate updating the relative numeric values. Also, relative numeric values should not be used with tags.

Example:

Tag References		Relative Numeric (Do Not Use)	
SAZ	MEM001-*-1	SAZ	3
INP	PTR004-*	INP	3
JMP*	MTA430	JMP*	*+4
LDA	LPA500	LDA	TAG+6
RAO	(COM250)	RAO	(TAG+5)
ADD-	BIT0	ADD-	$D4

H. MASTER CLEAR RESTART CONSIDERATIONS

1. Test will be designed to accomplish Master Clear Restart in the following sequence:

   a. Master Clear, set P register to the test initial address, and Run. IA contains a Jump (JMP*) instruction to the Parameter Entry Routine. This is to allow the operator to change parameters on restarting the test regardless of the condition of bit 10 Stop/Jump word.

   b. Subsequent to the execution of Parameter Entry Routine the test must request interrupt line from monitor, i.e. RTJ- (REQINT).

2. After completion of the above functions the test transfers control to the test section control routine and continues execution.

I. DESIGN CAUTION

Test will not be designed to have more than one level of indirect addressing due to 65K mode restrictions.

SUPPLEMENT E MONITOR BASED SUBROUTINES PROGRAMMING SPECIFICATION

I. INTRODUCTION

A. GENERAL

1. The collection of monitor based standardized subroutines, to be referred to as MBS, provides the basis for a pseudo language for diagnostics.

2. Using MBS results in the following:

   a) Tests are shorter; redundant coding is nearly eliminated.

   b) Tests are standardized; error reporting, except for data errors, is done through the monitor.

   c) The writing of tests is simplified giving the writer time to design a more thorough diagnostic.

3. MBS provides an environment for multiplexing tests. The multiplex philosophy is opposite standard SMM17 techniques in that it will try to trade off control every other time an I/O subroutine is called unless an interrupt condition exists or an inhibit trade flag (H0G) is set. Multiplexing tests this way results in trading off control at a much higher rate thereby more closely simulating 1700 MSOS.

B. 1700 MSOS FOREGROUND/BACKGROUND SIMULATION

1. During SMM17 initialization the protect option message will be output to the operator if the MBS bit was set in the SMM control word (bit 08).

2. In this mode the monitor is protected (foreground) and the tests are not protected (background).

3. Each MBS call results in a legal protect fault similar to 1700 MSOS. Non MBS protect faults will produce an error. This mode of operation is transparent to the diagnostic and does not impose special restrictions to the programmer.

   MBS tests cannot execute with standard SMM17 library tests in the Protect mode.

II. INTERFACING WITH MBS

A. TEST STRUCTURE

1. The ALF card at IA+1 must contain the letter P following the three letter test mnemonic. For example test 80 (BG504 drum test) would be:

```
ALF        3, DRMP80
```

2. IA+11 contains an ADC for the address of the error/operation file. This is a 93 word file that is located at the end of the test. It is constantly being updated by MBS with I/O data such as expected and actual status, type of operation being performed, and timing information. It also contains equipment codes and interrupt data. All MBS error information is pulled from the tests error file.

3. IA+12 describes the number of error files for the test. This number is typically zero for one file. If the test was talking to two devices, then IA+12 would be incremented by one prior to performing I/O on the second device. This process would continue for as many devices as required.

4. IA+13 contains the unit/station number when testing controllers that have sub-devices connected. This information will be output with the clear text message to be available on a future SMM17 release.

5. The error file has the following structure:

```
ERFILE     NUM       0          FWA of error file
           NUM       0          Status 1 address set by MBS
           NUM       $00XX      Status 2 director bits 0 = none
           NUM       $00YY      Status 3 director bits 0 = none
           NUM       $00ZZ      Status 4 director bits 0 = none
           NUM       0          Channel Address status set by MBS
           NUM       0          Equipment address set by MBS
           BSS       (TSDATA)   Remaining file data
```

The programmer must supply director bits or station codes necessary to copy status 2, 3, and 4 if they exist. The MBS initialize routine overlays the status 1 location with the address supplied to SMM17 at test load time. The remaining status addresses will be made if XX, YY, and ZZ are nonzero. The address to read channel address status will be set by MBS if a 1706/1716 was specified. The equipment number for the device will be set by MBS. The error file should be located at the end of the test. The value for TSDATA will be found in the EQU deck. Consecutive files must be located back-to-back.

B. AVAILABLE SUBROUTINES

1. The following subroutines are available for I/O:

   a. CKST - Check status

   b. RECKST - Recheck status

   c. MNTRST - Monitor status

   d. FN - Function

   e. RD - Read

   f. WR - Write

   g. RINT - Recognize interrupt

2. The following subroutines are available for utility:

   a. FIXDLY - Millisecond delay

   b. RDMLY - Random millisecond delay

3. The following subroutines are available for control:

   a. MSINIT - MBS initialize

   b. SETPP - Set protect bit

   c. CLRPP - Clear protect bits

   d. SELIN - Select interrupt

   e. DSELIN - Deselect interrupts

   f. HOG - Inhibit test multiplex

   g. ERROR - Report error

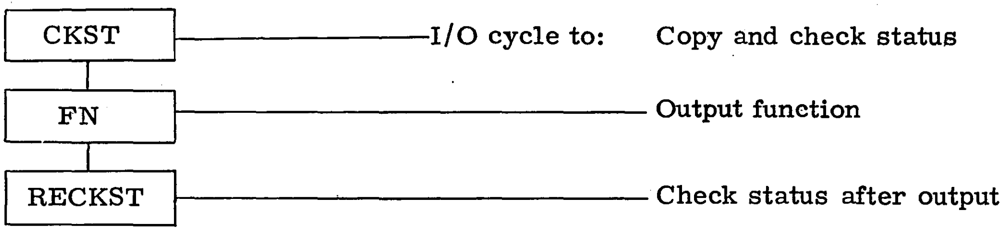
C. SUBROUTINE STRUCTURE

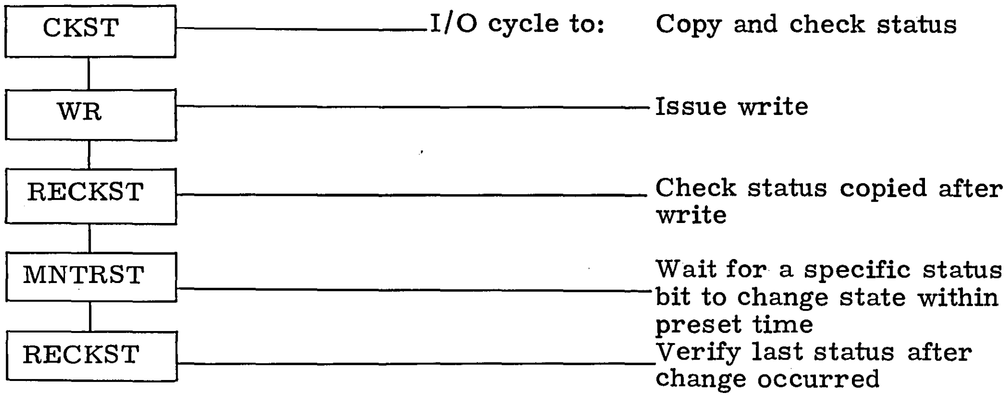
1. The most efficient use of the I/O subroutines is to group them in an I/O cycle.

2. An I/O cycle is the group of calls in the sequence required to issue a function, read or write to a specific device. Typically there is one I/O cycle per test but there is no limit. The I/O cycle is generally located near the test front end or before this first section.

3. The function of each section will be to set up the I/O cycle for the type of I/O required (FN, RD, WR). There is no limit of passes through the I/O cycle per section.

4. I/O cycle call examples:


CKST
I/O cycle to: Copy and check status
FN
Output function
RECKST
Check status after output


CKST
I/O cycle to: Copy and check status
WR
Issue write
RECKST
Check status copied after write
MNTRST
Wait for a specific status bit to change state within preset time
RECKST
Verify last status after change occurred

D. INTERRUPT PROCESSING

1. The condition of any given interrupt line can be in four modes:

   a. Illegal - This line has not been selected by a test.

   b. Expected - This line has been selected by a test, but the hardware interrupt has not occurred.

   c. Active - This line has received a hardware interrupt, it has been pre-processed by the monitor but it has not been processed by the test selected it.

   d. Process - This line has caused control to be given to the test that selected it and that test is now processing this interrupt.

2. A line is in the ILLEGAL mode if an interrupt occurs prior to selecting it or after the DSELIN call, deselect interrupt, has been issued.

3. MBS will handle a maximum of four lines per test. The input parameter for interrupt line assignment must have the following format:

B1	L1	B2	L2
B3	L3	B4	L4

Must be a two-word entry

B (X) = Bit position in director status word associated with assigned line.

L (X) = Assigned interrupt line.

EXAMPLE:

4500 0000	Controller has one interrupt on line 5 which correlates to status bit (E. O. P.).
3F4F 5F00	Status bits 3, 4 and 5 (data, E.O.P. and alarm) are on line 15.

4. The call format for DSELIN is:

CALL

```
           RTJ-       (DSELIN)
+0         NUM        $XXXX          X=bit number(s)
```

PARAMETERS

+0 Bit number(s)

The bit position of the expected interrupt in the equipment status word; must agree with interrupt input parameter(s).

RETURN

a) This interrupt line is now illegal.
b) The Interrupt Mark register for this bit is set.
c) The timer is cleared.

```
EXAMPLE    RTJ-       (DSELIN)
           NUM        $0018          Clear interrupt lines for
                                     data and E.O.P.
```

5. To make an interrupt(s) legal, the select interrupt call is used.

CALL

```
            RTJ-        (SELIN)
+0          NUM         $XXXX               X=Bit number(s).
+1          NUM         $tttt                t=System (not hardware)
                                               times.
+2          ADC         Processor address.
+3          ADC         Interrupt parameter FWA.
```

PARAMETERS

+0 Bit position in equipment status word (same as DSELIN).

+1 A maximum allowable time in milliseconds between Active and Process modes, max = 7FFF.

+2 Process routine address.

+3 First word address of two-word interrupt line assignment from parameters input area.

RETURN

Interrupts occurring will be in the Expected mode.

EXAMPLE

```
            RTJ-        (SELIN)
            NUM         $0010               Select line associated with
            NUM         $0100               bit 04.  Wait 256 milli-
            ADC         PROCES              seconds to process test.
            ADC         INTWRD              Process routine address
                                            first word of interrupt data.
```

6. The recognize interrupt routine is used by a diagnostic to process an expected interrupt.

CALL

```
                  RTJ-       (RINT)
          +0      NUM          *          Time Limit
          +1      NUM          *          Director 1 Status Mask
          +2      NUM          *          Director 1 Status Value
          +3      NUM          *          Channel Status Mask
          +4      NUM          *          Channel Status Value
          +5      NUM          *          Channel Address Mask
          +6      NUM          *          Channel Address Value
Optional {+7      NUM          *          Director 2 Status Mask
         {+8      NUM          *          Director 2 Status Value
          +9                              Error Return
          +10                             Normal Return
```

PARAMETERS

+0 Time Limit - Defines the maximum time allotted for the monitor to receive the interrupt. This time limit is to be in 1-millisecond increments, between the limits of 0 and 32,767 ($7FFF) milliseconds.

+1 +3 +5 +7 Status Mask - Defines those bits which are to be included in each status check while waiting for the interrupt to occur. Status is not copied or checked after the interrupt occurs.

+2 +4 +6 +8 Status Value - Defines the expected status value of masked status being checked.

+9 The monitor will return control to the diagnostic at +9 if any of the masked status is not exactly as specified by the respective status values, or the time limit specified in +0 has been exceeded. This must be a one-word instruction.

+10 When the selected interrupt occurs, the monitor returns control to the test's interrupt processor routine in the format of a return jump to the address specified in +2 of the select interrupt routine call. An indirect jump through this entry point to exit the processor routine will result in returning test execution to this point, the normal return at +10. For devices that do not have a second director status, error and normal return will be at +7 and +9.

NOTE

For equipments that will not be connected to a 1706/1716, the +3, +4, +5, and +6 parameters may be replaced with status 3 and 4 mask/value fields. Such devices connect directly on DSA.

EXAMPLE

```
        RTJ-    (RINT)
+0      NUM     10000     Wait 1 Second
+1      NUM     $FFFF     Status 1 Mask
+2      NUM     $0019     Status 1 Value
+3      NUM     $FDBF     Status 3 Mask
+4      NUM     $0021     Status 3 Value
+5      NUM     $FB7F     Status 4 Mask
+6      NUM     $0063     Status 4 Value
+7      NUM     $7FFE     Status 2 Mask
+8      NUM     $139C     Status 2 Value
+9      JMP*    ER8000    Error Return
+10     JMP     TAG       Continue at Normal Return
```

E. ERROR DETECTION - USE OF STATUS CHECKING

1. Error detection will be done in the monitor. The most often used routines will be those that check status, monitor status, or recheck status. As a result a diagnostic test should make efficient use of the status checking routines. If the reader refers to the description of the three types of status routines available, the following should be clear.

2. Each time a diagnostic performs a single operational or control function on a device, it is strongly recommended that it make use of all three status routines. In order to make use of these it is advisable that the test writer be certain of the equations for each bit in all status words. This information will serve the user in determining masks for the status routines and more important, in determining expected results from certain operations. The equations must be accurate and are made from the prints of a device and its controller.

3. It may also be helpful to list all computer originating function codes and to correlate those functions to the list of status bit equations. This list would then contain a function code followed by a set of bits which would be present if that function code were issued to a device.

4. With these two lists the diagnostic programmer should have the information necessary to make a call to any of the three status routines. The diagnostic should never check status without using a monitor routine. Before performing an operation, the diagnostic should call the check status routine to make sure the equipment can accept the operation. Next it should call the re-check status routine to determine the status that was copied immediately following the last operation. Finally, the diagnostic should call the monitor status routine which will continually check for a certain condition to be met, e.g., EOP either until that condition is met or a specified amount of time has passed. (Note that this time is specified by the diagnostic and it is up to the programmer to determine this time.) After performing all these status checks, there remains one final status check. The diagnostic should perform a RECKST after an MNTST to assure that the final status copied (just after the MNTST found the bit it was waiting for) is the one expected.

5. This series of status checks will provide the diagnostic with one of the most thorough means available to verify any operational or control function of a device. If the programmer ignores these routines, he will not have a good test under SMM17 3.0.

F. STATUS ROUTINES

1. Check Status

This routine copies and checks all the status of the device under test. The copied status is masked and compared to the expected values as specified by the call. Director 1 status, buffer channel status, and current channel address parameters must be included in the call. Director 2 status may be included if available from the device under test.

CALL

```
                  RTJ-       (CKST)
           +0     NUM          *          Director 1 Status Mask
           +1     NUM          *          Director 1 Status Value
           +2     NUM          *          Channel Status Mask
           +3     NUM          *          Channel Status Value
           +4     NUM          *          Channel Address Mask
           +5     NUM          *          Channel Address Value
Optional  {+6     NUM          *          Director 2 Status Mask
          {+7     NUM          *          Director 2 Status Value
           +8                             Error Return
           +9                             Normal Return
```

PARAMETERS

+0 +2
+4 +6 Status Mask - Defines those bits which are to be included in each status check.

+1 +3
+5 +7 Status Value - Defines the expected status value of each masked status being checked.

+8 Error Return - The monitor return control to the test at +8 if any of the masked status is not exactly as specified by the respective status values. The instruction located at CALLPT+8 must be a one-word instruction.

+9 The monitor returns control to the test at +9 if all the masked status values are as specified.

NOTE

For equipments that will not be connected to a 1706/1716, the +2, +3, +4, +5 parameters may be replaced with status 3 and 4 mask/value fields. Such devices connect directly on DSA.

EXAMPLE

Device has four levels of status and connect directly on DSA.

```
          RTJ         (CKST)
   +0     NUM         $DB7F        Status 1 Mask (ignore bits 7, 10,
                                   and 13)
   +1     NUM         $0019        Expected Value (EOP, data and ready)
   +2     NUM         $FFFF        Status 3 Mask
   +3     NUM         $1234        Expected Value
   +4     NUM         $FFFF        Status 4 Mask
   +5     NUM         $D636        Expected Value
   +6     NUM         $7FEO        Status 2 Mask (ignore bits 0-4, 15)
   +7     NUM         $0020        Expected Value
   +8     JMP*        ERE000       Error Return
   +9     JMP         CKE000       Normal Return
```

2. Monitor Status

This routine will copy and check all requested status while waiting for a specific status bit to change state within a specified time. The last status(s) copied which saw the bit change is not verified. To verify the last status copied the user must call the recheck status routine, RECKST.

CALL

```
                RTJ-        (MNTRST)

          +0    NUM         tttt         Time Limit
          +1    NUM         0CBS         Status Control Word
          +2    NUM         mmmm         Director 1 Status Mask
          +3    NUM         vvvv         Director 1 Status Value
          +4    NUM         mmmm         Channel Status Mask
          +5    NUM         vvvv         Channel Status Value
          +6    NUM         mmmm         Channel Address Mask
         ┌+7    NUM         vvvv         Channel Address Value
Optional <+8    NUM         mmmm         Director 2 Status Mask
         └+9    NUM         vvvv         Director 2 Status Value
          +10                            Error Return
          +11                            Normal Return
```

PARAMETERS

+0 Time in milliseconds to wait for condition to occur. Maximum time = $7FFF.

+1 Status Control word = 0CBS

15 9	8	7 4	3 0
	C	B	S

0 = Not used

C = Condition

1 = Wait for status bit to go off

0 = Wait for status bit to go on

B = Bit position in status word

S = Status word

0 = Director 1 Status

2 = Director 2 Status

4 = Channel Status (Director 3 Status)

7 = Channel Address Status (Director 4 Status)

+2, +4
+6, +8 Status Mask - Defines those bits which are to be included in each status mask.

+3, +5
+7, +9 Status Value - Defines the expected status of each masked status being checked.

If a 1706/1716 will not be used, then +4, +5, +6 may be replaced with director 3 and 4 status if needed.

+10 Error Return - This must be a one-word instruction. Control is returned here if:

a. A status error was detected while waiting for the condition to be met.

b. The condition was not met within the time specified.

+11 Normal Return - There were no status errors and the condition was met within the specified time.

EXAMPLE

Device does not have a director 2 status.

```
          RTJ-       (MNTRST)
+0        NUM        1000         Wait 1 second for bit 01
+1        NUM        $0110        to go off in status word 1 (busy)
+2        NUM        $FDFF        Status 1 mask, ignore bit 09
+3        NUM        $000D        value, expect ready, busy, data
+4        NUM        $FFFF        channel status mask
+5        NUM        $0011        value, expect ready, EOP
+6        NUM        $FFFF        channel address mask
+7        NUM        $13A6        value of expected address,
+8        JMP*       ERE000       error return,
+9        JMP*       CKE000       normal return.  Check repeat
                                  conditions.
```

3. Recheck Status

This routine checks the status previously copied by another routine. The purpose is to verify the final status copied prior to exiting the Read, Write, Function, or Monitor Status routines. With the exception of not copying new status, this routine is identical to the check status routine.

CALL

```
            RTJ-      (RECKST)
     +0     NUM       *              Director 1 Status Mask
     +1     NUM       *              Director 1 Status Value
     +2     NUM       *              Channel Status Mask
     +3     NUM       *              Channel Status Value
     +4     NUM       *              Channel Address Mask
     +5     NUM       *              Channel Address Value
     +6     NUM       *              Director 2 Status Mask
     +7     NUM       *              Director 2 Status Value
     +8                              Error Return
     +9                              Normal Return
```

PARAMETERS

+0 +2
+4 +6 Status Mask - Defines those bits which are to be included in each status check.

+1 +3
+5 +7 Status Value - Defines the expected status value of each masked status being checked.

+8 The monitor return control to the test at +8 if any of the masked status is not exactly as specified by the respective status values. The instruction located at +8 must be a one-word instruction.

+9 The monitor returns control to the test at +9 if all the masked status values are as specified.

NOTE

For equipments that will not be connected to a 1706/1716, the +2, +3, +4, and +5 parameters may be replaced with status 3 and 4 mask/value fields. Such devices connect directly on DSA.

G. FUNCTIONS I/O

1. The actual exercising of a device is accomplished through the use of the Function routine and Read and Write routines. With status checking and use of this Function routine it is possible to perform all related device operations and check all the responses. Errors are automatically picked up by the monitor. The diagnostic need only attempt all the functions. It is up to the diagnostic programmer to determine any timing considerations, necessary function calls, any peculiar orders of functioning which may cause an error, or any other device manipulation.

2. The function routine issues the specified function code, copies all the available status* and branches on the specified response control depending on reject and reply.

3. FN (Function)

   This routine issues the specified function code, copies all the available status and branches on the specified response control depending on reject or reply. The following flow chart should be referenced for more detail on FN operations.

   CALL

```
                RTJ-          (FN)
       +0       NUM           ffff         f  =  Function Code
       +1       NUM           rrdd         r  =  Response Control/d = Director Bits
       +2       NUM           tttt         t  =  Timer
       +3       NUM           bbbb         b  =  17X6
       +4       Error Return
       +5       Normal Return
```

PARAMETERS

+0 Function code

The actual function word (16 bits) to be issued to the equipment.

+1 Response Control/D

15 8		6 0
Response Control	▨	D

Response control; the specified action to be taken after a reply or reject.

00 = CNT
Continue, regardless of response

01 = ERP
Error on reply
Continue on reject

02 = HRP
Hang on reply
Continue on reject

10 = ERJ
Error on reject
Continue on reply

20 = HRJ
Hang on reject
Continue on reply

22 = HNG
Hang, regardless of response

NOTE

Bit 15 set in the RRDD word negates copy status and should only be used for special hardware.

D = director bits; appropriate director bits for equipment address to perform desired function.

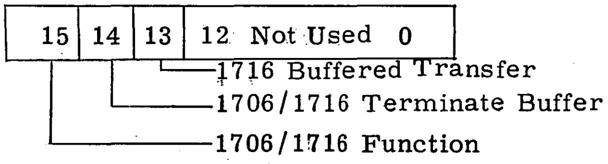
+2 Timer

The maximum time (in milliseconds) to hang (HNG, HRP, HRJ).

When the timer has expired, an error will occur.

+3 17X6

This parameter indicates a function request to a 1706 or 1716.



RETURN

There are two returns:

+4 Indicates the monitor has sensed an error (see ERR)

+5 Normal return

TIMING

Assuming no memory cycle stealing due to in process DSA transfers the function requires 400 microseconds on a 1704.

4. The read routine transfers data from the device to the CPU using A/Q or DSA. The routine will branch according to the response (reject, reply) and the specified response control.

Buffered I/O - The response is checked only on the output to the DSA channel. Control is returned to the caller immediately.

A/QI/O - The response is checked after every transfer. Control is returned to the caller after the specified number of words have been transferred.

CALL

	RTJ-	(RD)	
+0	NUM	TTTT	Transfer type
+1	NUM	1111	No. of words
+2	ADC	aaaa	Data buffer FWA
+3	NUM	caca	Continue bit first channel address
+4	NUM	cccc	No. of channels
+5	NUM	RRDD	Response control/director bits
+6	NUM	tttt	Timer (millisecond)
+7			System overload return
+8			Error return
+9			Normal return

PARAMETERS

+0 Transfer type

0 = AQ

1 = DSA

To initiate DSA transfers, the FWA of the buffer (from +2) is output to the device.

+1 Number of words to be transferred

+2 First word address of data buffer

+3 First extended address (hardware) for device using continue bit addressing (bit 15)

+4 Number of continue bit addresses to increment

+5 Response control

15	13		8	6 0
C	▨	RR	▨	DD

DD = Director (station) bit positions to perform read

RR = Response control

10 = Continue on reply, error on external reject

01 = Continue on external reject, error on reply

20 = Continue on reply, repeat for external reject

C = Copy status after read

0 = Copy status (all specified levels)

1 = Do not copy status. The do not copy feature is for special hardware applications only and is not recommended for standard 1700 peripherals.

+6 Time in milliseconds to wait for a reply as defined in the response control code RR=20.

+7 The read routine will return here if this call or another test has made an A/Q read request while a DSA read is in progress. This must be a one-word instruction.

+8 Control is returned here if:

a. A response other than expected occurred.

b. The timer expired while waiting for a reply response.

The error return must be a one-word instruction.

+9 Control is returned at the normal return after:

a. All A/Q read transfers are complete.

b. One output to start DSA transfers

TIMING

From	To	Time
+0 transfer	First transfer	320 microseconds
Nth transfer	N +1 transfer	30 microseconds
Last transfer	First copy status	100 microseconds
Last copy status	+9 return	33 microseconds

NOTE

If DSA transfers were initiated, the caller must use the monitor status call before exiting the I/O cycle for this call.

5. The write routine transfers data from the CPU to the device using A/Q or DSA. The routine will branch according to the response (reply, reject) and the specified response control.

Buffered I/O - The response is checked only on the output to the DSA channel. Control is returned to the caller immediately.

A/Q I/O - The response is checked after every transfer. Control is returned to the caller after the specified number of words have been transferred.

CALL

	RTJ-	(WR)	
+0	NUM	TTTT	Transfer type
+1	NUM	llll	No. of words
+2	NUM	aaaa	Data buffer FWA
+3	NUM	caca	Continue bit first channel address
+4	NUM	cccc	No. of channels
+5	NUM	RRDD	Response control/director bits
+6	NUM	tttt	Timer (millisecond)
+7			System overload return
+8			Error return
+9			Normal return

PARAMETERS

+0 Transfer type

0 = A/Q
1 = DSA

To initiate DSA transfers, the FWA of the buffer (from +2) is output to the device.

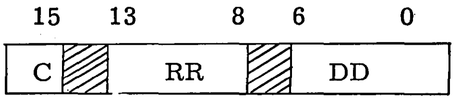
+1 Number of words to be transferred.

+2 First word address of data buffer.

+3 First extended address (hardware) for device using continue bit addressing (bit 15).

+4 Number of continue bit addresses to increment.

+5 Response control.



DD = Director (station) bit positions to perform write
RR = Response control

10 = Continue on reply, error on external reject
01 = Continue on external reject, error on reply
20 = Continue on reply, repeat for external reject

C = Copy status after read

0 = Copy status (all specified levels)
1 = Do not copy status. The do not copy feature is for special hardware applications only and is not recommended for standard 1700 peripherals.

+6 Time in milliseconds to wait for a reply as defined in the response control code RR=20.

+7 The read routine will return here if this call or another test has made an A/Q read request while a DSA read is in progress. This must be a one-word instruction.

+8 Control is returned here if:

a. A response other than expected occurred.
b. The timer expired while waiting for a reply response.

The error return must be a one-word instruction.

+9 Control is returned at the normal return after:

a. All A/Q read transfers are complete.
b. One output to start DSA transfers.

TIMING

From	To	Time
+0	First transfer	320 microseconds
Nth transfer	N +1 transfer	30 microseconds
Last transfer	First copy status	100 microseconds
Last copy status	+9 return	33 microseconds

NOTE

If DSA transfers were initiated, the caller must use the monitor status call before exiting the I/O cycle for this call.

EXAMPLE:

```
REPEAT
               RTJ-     (RD)
     +0        NUM      $0000       A/Q transfer
     +1        NUM      $0100       Transfer length = 256
     +2        ADC      RDBUF       FWA of read buffer
     +3        NUM      $9400       First continue bit address
     +4        NUM      0008        Number of address increments
     +5        NUM      $1001       Continue on reply error on reject,
                                    dir = 01
     +6        NUM      $0000       millisecond timer = 0
     +7        JMP*     REPEAT      System busy return
     +8        JMP*     ERE000      Error return
     +9        JMP      CKE000      Normal return

REPEAT
               RTJ-     (RD)
     +0        NUM      $0000       A/Q transfer
     +1        NUM      $0040       Transfer length = 64 words
     +2        ADC      RDBUF       FWA of read buffer
     +3        NUM      0           Not used
     +4        NUM      0           Not used
     +5        NUM      $2000       Response control, director = 00
     +6        NUM      0           Timer = 0
     +7        JMP*     REPEAT      System busy return
     +8        JMP*     GRE000      Error return
     +9        JMP*     CKE000      Normal return
```

III. ERROR REPORTING

A. ERROR HANDLING

1. When a monitor routine senses an error, control is returned to the caller at the error return address. On the return, the A register will contain an error code describing the type of error encountered.

B. ERROR CODES

1.

Subroutine	Error Code
a. CKST	0003
b. MNTRST	0000, 0003
c. RECKST	0003
d. FN	0001 0002
e. RD	0001 0002
f. WR	0001 0002
g. RINT	0003, 0004

2. Error Code Definitions

	MBS Error Code	Definition
a.	0000	Monitor status time out
b.	0001	I/O response time out (reply, reject)
c.	0002	Response was other than predicted
d.	0003	Status error actual ≠ expected
e.	0004	Interrupt time out. Interrupt did not occur within specified time

3. The error subroutine gets the data from the error file and outputs the error.

CALL

```
                RTJ-     (ERROR)
    +0          NUM      $SSEE          Section/error code
    +1          ADC      RRRR           Repeat conditions address
    +2             Normal return        No repeat requested, continue
```

PARAMETERS

+0 SS = Section code
EE = Error code

+1 Address to repeat conditions if operator requested

+2 Control returned here if repeat conditions not requested

EXAMPLE

```
                RTJ-     (ERROR)
    +0          NUM      $0630          Section 6, error 30
    +1          ADC      RPE000         Repeat conditions address
    +2          JMP*     EXIT           Return to test section
```

4. Data errors are not handled by MBS. The user must use standard SMM17 error reporting techniques for data errors.

5. I/O cycle error reporting is as follows:

The first four pairs of A/Q typeouts have a standardized definition. These definitions are as follows:

A1	Q1	A2	Q2	A3	Q3	A4	Q4
ID	ST/JP	SS/EE	RTN	MBSERR	IO/RP	(A)	(Q)

ID = SMM ID word
ST/JP = SMM Stop/Jump parameter word
SS/EE = Test section number and test error number
RTN = Return address
MBSERR = MBS monitor detected error code

0000 - Status time out error
0001 - I/O time out error
0002 - I/O response error
0003 - Status error
0004 - Interrupt time out error

IO/RP = Last I/O operations performed and the associated response

IO - 10 = Write
20 = Read
30 = Function

RP - 10 = Reply
20 = External Reject
30 = Internal Reject

(A), (Q) = Register contents relative to the last I/O operation.

The next three pairs of A/Q typeouts vary with the type of MBS monitor detected error code. The content of these three pairs of typeouts are as follows:

MBS Error Code	A5	Q5	A6	Q6	A7	Q7
00	ACT STS1	ACT STS2	CLOCK	OCBS	-	-
01	ACT STS1	ACT STS2	CLOCK	-	-	-
02	ACT STS1	ACT STS2	-	-	-	-
03	ACT STS1	EXP STS1	ACT STS2	EXP STS2	-	-
04	ACT STS1	ACT STS2	CLOCK	INT LINE	INT MASK	ASGN LINES

ACT STS1, ACT STS2 - Actual status 1 and status 2 copied from the device.

EXP STS1, EXP STS2 - Expected status 1 and status 2.

CLOCK - The time limit in milliseconds which expired waiting for the specified condition to occur.

OCBS - The control word being used during monitor status (MNTRST).

INT LINE - The line number of the interrupt which occurred.

INT MASK - The interrupt mask being used at the time of the interrupt.

ASGN LINES - A summation of all the interrupt lines assigned to the device, i.e. 0050 = interrupt line 5.

EXAMPLE

A1	Q1	A2	Q2	A3	Q3	A4	Q4	A5	Q5
8068	0241	0630	12D5	0003	3010	0001	0381	0218	0219

Error analysis:

1. Test 80
2. Section 6, error code 30
3. Return address = $12D5
4. MBS error code = 0003, status error
5. Last operation performed was a function, response was a reply
6. Contents of the A register = 0001
7. Contents of the Q register = 0371
8. Actual status was $0218
9. Expected status was $0219

(The device dropped bit 00 (ready) after a clear controller.)

IV. CONTROL

A. MULTIPLEX CONTROL (MPX)

1. The purpose of this routine is to provide multiplexing control between tests. This subroutine is not available to tests, however, control is provided with the hog subroutine.

2. The MPX subroutine is called by all status and I/O routines except recognize interrupt (RINT). If the hog flag was set previous to the call, control is returned to the I/O or status routine in 15.4 microseconds. See Figure 1.

3. The set hog format is as follows:

CALL

```
                 RTJ-                       (HOG)
     +0                      Return
```

PARAMETERS

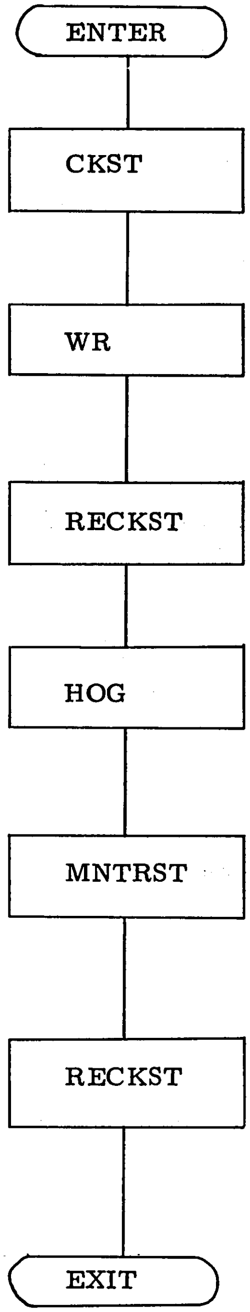
None

RETURN

Hog flags 1 and 2 are set. I/O or status call is guaranteed. Hog is cleared prior to executing the request, therefore, it must be set each time it is required. Interrupts to non MBS tests will be stacked until next call to MPX.

EXAMPLE

To ensure that the MNTRST call is performed in the following I/O cycle:



Copy and check status.

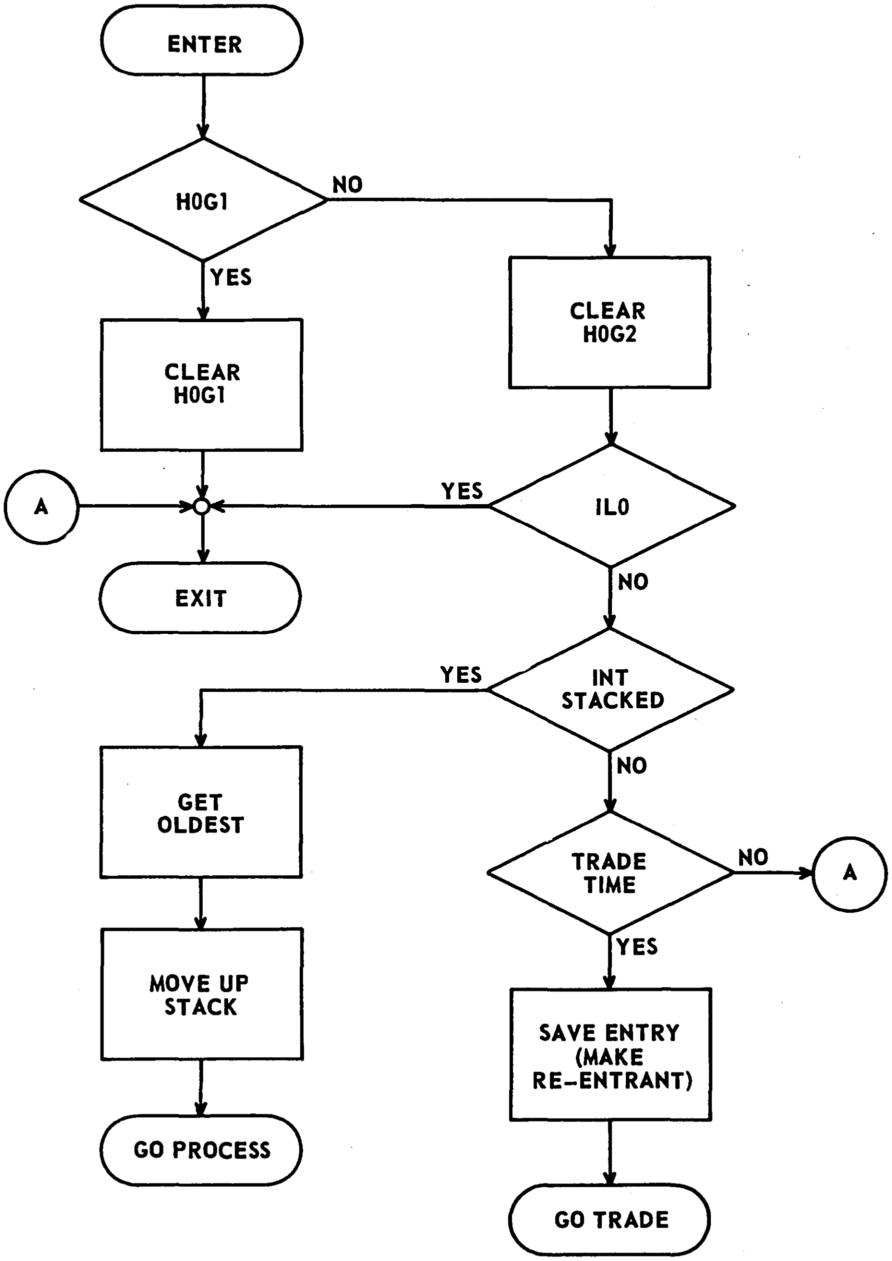
Issue write, copy status.

Check status after write.

Do not trade off control.

Copy and check status wait for specified status bit to change state.

Check last status copied when status bit changed state.



H0G1	Do not release control.
H0G2	Stack non MBS interrupts until next MPX call.
IL0	Interrupts in PROCESS mode (Interrupt Lock Out).
TRADE	Release control to monitor every other call.

Figure 1.

B. SET PROTECT BITS

1. The set protect bit routine is used to set memory protect in the callers test area. Care must be used to ensure protect bits are not altered in the monitor.

CALL

```
             RTJ-      (SETPP)
    +0       ADC       fwa
    +1       NUM       1111
    +2            Return
```

PARAMETERS

+0	First word address of protect area
+1	Number of words to protect

RETURN

Protect bits are set in callers protect area

C. CLEAR PROTECT BITS

1. The clear protect bit routine is used to clear memory protect bits in the callers test area. Care must be used to ensure protect bits are not altered in the monitor.

CALL

```
             RTJ-      (CLRPP)
    +0       ADC       fwa
    +1       NUM       1111
    +2            Return
```

PARAMETERS

+0	First word address of protected area
+1	Number of words to clear

RETURN

Protect bits are cleared in callers protected area.

V. UTILITY

A. DELAY

1. CALL

```
              RTJ      (FIXDLY)
    +0        NUM      tttt          TIMER
    +1        Return
```

PARAMETERS

+0 tttt
Time in milliseconds. Range = 0 to 7FFF.

+1 Return after delay.

B. RANDOM DELAY

1. Minimum and maximum values in milliseconds are given. The delay will range somewhere between these two values.

CALL

```
              RTJ-     (RDMDLY)
    +0        NUM      xxxx
    +1        NUM      yyyy
    +2        Return
```

PARAMETERS

+0 xxxx
Minimum value in milliseconds.

+1 yyyy
Maximum value in milliseconds.

+2 Return after delay.

PRINTER/TELETYPE DUMP

(DMPA3B Test No. 3B)

I. INTRODUCTION

The Dump routine is a service routine for obtaining memory dumps on a line printer or a teletype. This routine is intended for use by customer engineers for dumping SMM test buffer areas during troubleshooting, and also to dump all of memory (if using a printer) when reporting diagnostic problems by DPSR. Snap dump and breakpoint dump options are also included for diagnostic debugging.

II. REQUIREMENTS

A. HARDWARE REQUIREMENTS

1. A 1700 series computer
2. One of the following:
   a. 1742 Line Printer
   b. 1740-501 Line Printer
   c. 9322/9323 Line Printer
   d. 1711/1712/1713 Teletype

B. SOFTWARE REQUIREMENTS

1. Dump loads under control of SMM17.
2. Dump does not take control from the monitor at any time. The operator must give the routine control by starting from its initial address (IA).
3. Dump is less than 200_{16} locations long.

III. OPERATIONAL PROCEDURE

A. LOADING PROCEDURE

The routine must be loaded under SMM17 as test number 3B. The selected equipment address determines which device type will be used. An equipment address of 0091 selects the teletype. Setting bit 15 of the equipment address selects a 9322/9323. All other equipment addresses select a 1742/1740-501. After being loaded, this routine repeatedly executes test exit monitor calls until the frequency (FC) equals zero and the routine is removed from the SMM test list. To now execute the dump program, it must be started at its initial address (IA).

B. PARAMETERS

1. Forcing a Parameter Stop

A parameter stop in Dump is not determined by the Stop/Jump word. A parameter stop will always occur when the routine is started at IA by the operator. There are two methods of forcing a parameter stop to start the routine.

a. If the contents of registers are not important at the time of entering Dump:

1) Master/Clear.

2) Select P register and set P=IA of the routine.

3) Run.

b. If the contents of the registers are important at the time of entering Dump:

1) Depress STEP key.

2) Continue depressing STEP key until INSTRUCTION light comes on.

3) Select and clear X register.

4) Select and clear P register.

5) Set P=IA of the routine.

6) Run.

2. Selecting Parameters

When the routine is started at IA, the following parameter stops will occur.

A1 = 3B21 ID word

Q1 = 0000 or Snap address (Biased address, not listing)

A1/Q1 are used to select one of three dumping modes:

Panic dump, Snap dump, or Breakpoint dump.

All three modes also contain an A2/Q2 parameter stop (see next page).

Panic dump is selected when Q1 is set to 0000. In this dumping mode the dump is initiated immediately after the parameter stop. When the dump is complete, the parameter stop will again occur. Any dumping mode may then be selected.

Snap dump is selected when Q1 is set to a non-zero snap address and A1 is set to 0000. In this dumping mode a dump is initiated for each time the instruction at the snap address is executed. The dumps will be enabled until a new dump is selected by forcing a dump routine parameter stop. If an illegal snap address is selected, the A1/Q1 stop will be repeated with Q1=0000. Refer to Snap/Breakpoint dump "caution", Section V.

Breakpoint dump is selected when Q1 is set to a non-zero snap (breakpoint) address and A1 is not altered. In this dumping mode a dump is initiated for each time the instruction at the snap address is executed. After each dump, a dump routine parameter stop occurs. Any dumping mode may then be selected. If a panic dump is selected, A1 should be unaltered to enable another parameter stop after the dump is complete. If a snap or breakpoint dump is selected, a new snap address may be selected. A different area of memory to be dumped may be selected with any of the dumping modes. Once a Snap dump is selected, a Breakpoint dump can only be reselected by forcing a dump routine parameter stop. If an illegal Snap address is selected, the A1/Q1 stop will be repeated with Q1=0000. Refer to Snap/Breakpoint dump "caution", Section V.

A2 = First Word address of memory area to dump

Q2 = Last Word address of memory area to dump

If Q2 is set to 0000, no memory dump will occur. Only the contents of the registers A, Q, I, and M will be dumped.

C. SNAP/BREAKPOINT DUMP INFORMATION STOP

This stop occurs to signal the operator to start the test in which the Snap address resides.

A = Instruction at the Snap address

Q = Snap address

This stop will occur only if a Snap dump or a Breakpoint dump has been selected by forcing a dump routine parameter stop (running from the dump routine's IA). The contents of A and Q are not parameters, but rather for information only. When this stop occurs, the operator must start the test in which the Snap address resides. The common method to do this would be to restart that test at its initial address (IA).

IV. OPERATOR COMMUNICATION

A. NORMAL MESSAGES/STOPS

When the routine is loaded, the following will be typed:

```
DMPA3B PRINTER/TTY DUMP
    Q1 = 0 OR SNAP ADR
    A2/Q2 = FWA/LWA
    IA = XXXX
```

B. ERROR MESSAGES/STOPS

If the selected Snap address contains an instruction considered illegal to this routine, the A1/Q1 parameter stop occurs again. The Snap address will have been cleared in Q1. No other indication of an illegal request will occur. Refer to Snap/Breakpoint dump "caution", Section V.

V. DESCRIPTION

A. GENERAL

Dump is a service routine which is used for obtaining memory dumps on a line printer or a teletype. The routine does not use any of the monitor routines after it has been loaded and initialized. This is to prevent changes from being made to memory areas that may be dumped.

B. PANIC DUMP

The Panic dump does an immediate dump of the operational registers (A, Q, I, and M) and a selected area of memory. The operational register contents are saved if the dump program is entered by stepping the computer to the start of an instruction cycle and then setting the P register to the dump program's IA. Master clearing the computer and setting P will, of course, clear the operational registers. Under this dumping mode P (Snap address) will always be dumped as 0000.

C. SNAP/BREAKPOINT DUMP

The Snap/Breakpoint dump does conditional dumps of the operational registers (P, A, Q, I, M) and a selected area of memory. The P register dump reflects the selected Snap address. The selected Snap address determines when the dumps will occur. During the parameter stop, the instruction at the Snap address is saved and a return jump to a Snap/Breakpoint dump processor is stored in its place. When the program being snapped reaches the Snap address, the return jump will give control to the Snap/Breakpoint dump processor rather than execute the instruction that was there previously. The Snap/Breakpoint dump processor first inhibits interrupts and saves the operational registers so that they can be restored before returning to the program being snapped. Then the dump of the registers and memory take place. If the selected dump is a Breakpoint dump, a dump routine parameter stop occurs. In either dumping mode, the registers are then restored and the instruction replaced by the return jump is executed. Interrupts are enabled just before exiting the Snap/Breakpoint processor. The exit is to the snapped program at its Snap address plus one. The Snap/Breakpoint dumps will continue until the program being snapped is halted or the return jump at the Snap address is no longer being reached. A new dump may be selected by forcing a dump routine parameter stop. When a new Snap address is selected, the instruction for the previous Snap address will be restored if that snapped program has not been overlayed with a new program. (The previous Snap address is checked for a return jump to the Snap/Breakpoint dump processor.)

CAUTION

Care must be taken in selecting a Snap address. The following restrictions must be considered:

1. During the snap/breakpoint dump processing, an interrupt cannot be pending nor can an interrupt be in process. A pending interrupt would be delayed until the Snap/Breakpoint dump was complete. The same would be true of an interrupt being processed.

2. During Snap/Breakpoint dump processing, any timing operation would become grossly inaccurate.

3. The instruction at the Snap address is limited by all of the restrictions as follows:

    a. Two-word storage reference instructions are illegal (DELTA=0).

    b. Relative addressing storage reference instructions are illegal (Bit 11=1).

    c. Return Jump instructions are illegal (F=5).

    d. Register reference instructions SLS, INP, OUT, EIN, IIN, and all Skip instructions are illegal (F1=0-5).

TAPE TO PRINT ROUTINE

(LST03C Test No. 3C)

INTRODUCTION

This routine accepts BCD magnetic tape information from a 1731 or 1732 Controller and lists it on a 1740/501 or a 1742 Printer.

OPERATIONAL PROCEDURE

A. HARDWARE REQUIREMENTS (Minimum)

1. One 17X4 Computer with 4K of memory.
2. One 1731/601 or one 1732/608-609 tape system.
3. One 1742 printer or one 1740/501 printer.

B. SOFTWARE REQUIREMENTS

1. LST routine must be loaded under control of SMM 1700 System Maintenance Monitor.
2. Upon completion of loading, the following typeouts occur:

```
$3C *LIST UNIT 6*
IA= XXX, FC=XX
(AU)   = TAPE EQ
(AL)   = PRINT EQ
(Q)    = # FILES/SKIP             (number of files to skip before
(M)    = 0 *PROG CONT*            listing)
(M)    ≠ 0 *SINGLE SP*
LISTS   1 FILE
```

3. Computer will stop after loading to inspect and/or change (A)*, (Q) and (M).
4. LST routine will list unit 6 until a file mark is detected and then halt. Running from this point will list the next file on tape.
5. LST routine can be re-started by starting at IA.
6. All necessary operating instructions are contained in the typeout at load time.

*Note: (A) Indicates the contents of register

3000 CHANNEL SIMULATOR ASSEMBLER
(SAS03D Test No. 3D)

I. OPERATIONAL PROCEDURE (see SMM Programming·Reference Manual for a more detailed procedure.)

A. RESTRICTIONS

1. Runs on a 1700 system with the following minimum configurations.

   a. Tape units (2 each)

      1) Equipment #7
      2) Interrupt line #3
      3) Converter #0
      4) Source unit #0
      5) Object unit #1

   b. Line printer on equipment F

   c. Teletype

   d. Core size of 8K

2. Object wrap-around after 256 instructions (W error).

3. Object is assembled in absolute format.

4. Object is intended to be written on 7 track tape.

5. A maximum of 256 symbols may be used.

6. A maximum of 56 remark cards may used (an * in column 1).

7. A maximum of 7 equate cards are allowed for each 40_8 instruction entered.

8. SAS overlays the SMM17 monitor.

B. LOADING PROCEDURE

1. Call as standard SMM17 test #3D.

C. PARAMETERS

1. SKIP switch set inhibits the writing of a file mark after the object.

2. If the SELECTIVE STOP is set, the program stops have the following meaning:

   a. Stop 1. = Tape assignment with A =

```
              3070   Altering A changes the assignment.
INT. line------┘|└---Equipment No.
     Conv. No.--┘
```

b. Stop 2. A = 0. Set A to move the source tape A files forward before assembly.

c. Stop 3. A = 0. Set A to move the object tape A files forward before assembly.

II. MESSAGES

A. NORMAL

1. SIMASSEM
   3000 CH. SIMULATOR ASSEMBLER
   Notes beginning of assembler.

2. Name =

   Requests the programmers name. Up to 22 characters may be entered. 22 characters or a carriage return causes the program to continue.

B. ERROR CODES

1. S = Symbol assignment impossible (symbol value forced = 0).
2. D = Symbol in source deck 2 or more times (value forced = 0).
3. A = Address magnitude or assignment is impossible (value forced = 0).
4. O = Illegal Op code. If possible, the address field will be defined and used as if a "VAL" type instruction were given.
5. U = The symbol being referenced is undefined. The address is given the value of 0.
6. W = Simulator core wrap-around. Over 256 instructions have been generated for the object and actual object wrap-around is occurring.
7. I = Ident error. Either the Ident is not present or it is not the first (non REM) card present in the source deck.

III. PROGRAM DESCRIPTION

A. GENERAL

The programmer writes machine language in mnemonic and symbolic form, specific constants and pseudo instructions.

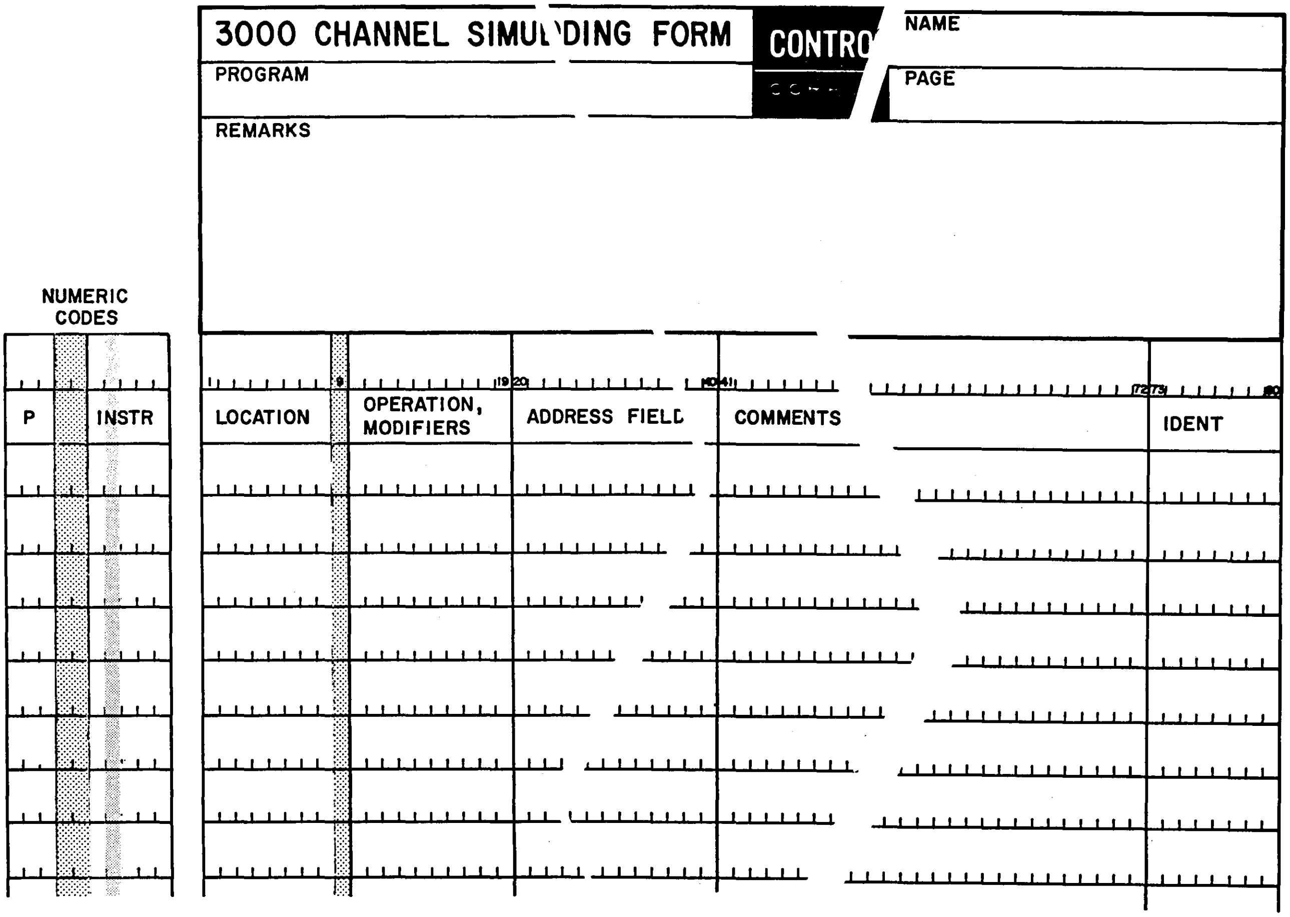
A program is generated between the IDENT pseudo instruction and END pseudo instruction.

The assembler assumes the program begins at location 0. Locations are assigned sequentially from 0 unless a PURG pseudo is encountered. To be legal, this instruction must move the program count forward by at least one location. If so, the program count is then advanced to the specified point and sequential operation continues from there.

B. INSTRUCTION FORMAT

This is a fixed format assembler; i.e. the position of the data in the record is essential for proper operation.

From the coding form, cards are punched and eventually placed on magnetic tape for use by the assembler. This tape is called the source tape and contains 80 character (BCD) records for each line entered. The correspondence between the columns on the coding sheet, card, and magnetic tape record are 1 to 1.



3000 CHANNEL SIMUL'DING FORM

CONTRO

NAME

PROGRAM

PAGE

REMARKS

NUMERIC CODES

P	INSTR	LOCATION	OPERATION, MODIFIERS	ADDRESS FIELC	COMMENTS	IDENT
		1	9	19 20	40 41	72 73 80

Each line of code has the following 5 fields; all instructions are defined in terms of the contents of these fields in the appendix.

1.	Location	Columns 1-8 (9 is always blank).
2.	Operation	Begins in column 10 and proceeds through column 18.
3.	Address	Begins in column 20 and proceeds until first blank.
4.	Comments	Normally begins in column 41 but may begin after first blank in address field.
5.	Sequence No.	A unique set of alphanumeric characters used to aid in editing. No rules or restrictions are placed on these characters by the assembler. They must follow the rules set by the edit routine used for updating. Card columns 73-80.

C. CARD IMAGE RULES

1. Columns 1-8 (SYMBOL)

   a. Up to 8 characters, beginning in column 1, are allowed. Do not include:

      1) The symbols +, -, or *.

      2) Imbedded spaces.

2. Column 1 = an * (REMARKS CARD)

   a. These cards are saved from pass 1 for printing on pass 2. A maximum of 56 remark cards are allowed for a full 8 page assembly.

3. Column 20 (ADDRESS)

   a. With the exception of BCD entries, the address field will terminate on the first blank after column 20. The following symbols may be used.

      1) + add this value to the preceding value.

      2) - subtract this value from the preceding value.

      3) * the value of this address.
         ** gives twice this address, not zero.

   b. Any combination of values may be used so long as the computed end result does not exceed the following limitations.

1) 255 for addresses and enters.

2) 4095 for constants.

3) + or - 11 for shifts.

4) 63 for increase and decrease "A".

4. Column 41 (COMMENTS)

There are no restrictions as to the content of the comments field.

3000 CHANNEL SIMULATOR INSTRUCTION SET

Instruction	Address	Remarks
CAA	D*	Clear alarm light A.
CAB	D*	Clear alarm light B.
CLA	D*	Clear A.
CLSO	D*	Clear special operation.
CON	D*	Connect with code in A.
CONC	D	Connect continuous, code in A. See D values.
COPY	D*	Copy status to A/.
CPA	M	Skip if (A) = (M).
CPS	M	Skip if (A) Status = M.
CRL	D*	Clear read line.
CRP	D*	Clear rej. /par. indications.
CWL	D*	Clear write line.
EINT	D*	Enable interrupt.
ENA	Y	Enter A with Y. Y = 0-256.
ICA	D	Input continuous to A. See D values.
INA+	Y	Increase A +1 Y times. Y = 0-63.
INA-	Y	Increase A -1 Y times. Y = 0-63.
INAW	D*	Input to A.
INCL	D*	Clear interrupt.
INPW	M, N	Input N words to M M+N.
IOCL	D*	Clear I/O.
LAA	D*	Light alert lamp A.
LAB	D*	Light alert lamp B.
LDA	M	Load A from M.
NEG	D*	Negate BCD conversion.
NOP	D*	No operation.

M = Memory location (000 377_8) N = word count to A (000 377_8).

Y = Constant (value varies with instruction using it).

D* = 00 17_8 which will be placed in the instruction but is not used by this instruction.

Instruction	Address	Remarks
OCA	D	Output continuous from A/.
OTAW	D*	Output word from A.
OUTW	M, N	Output from M M+N, N words.
PEJ	M	Jump to M if a parity error.
RTJ	M	Jump to M with jump to P=1 in A/.
SEL	D*	Select with function in A.
SELC	D	Select continuous with function in A. (see D values).
SET	D*	Set A to all 1's.
SHA	+Y	Shift (A) left 1 enough times to accomplish +Y left end around or -Y right end around. Y = 1 11.
SJA	M	Jump to M if JUMP switch A is set.
SJB	M	Jump to M if JUMP switch B is set.
SOF	D*	Turn sound alarm off.
SON	D*	Turn sound alarm on.
SRL	D*	Set read line.
STA	M	Store A at M.
SUP	D*	Suppress assembly/disassembly.
SWL	D*	Set write line.
TCA	D*	A A
TDA	D*	Transfer data switches to A.
TEA	D*	Transfer Equipment switch to A upper.
TPA	D*	Transfer P+1+jump to A.
UCS	D*	Unconditional stop.
UJP	M	Unconditional jump to M.
WMO	D*	Select Word Mark Operation.

Assembler Pseudo Op. codes

BCD	N, CCCCC...	Enter N locations with BCD characters C. N = 1 29.
BSS	N	Clear and save N locations. N = 1 256.
END		End of assembly.
EQU	X	Equate the symbol (column 1 8) to X.

Instruction	Address	Remarks
IDENT	NAME	Begin assembly card. Names program (up to 8 characters). Columns 41 through 59 (19 characters for data and version) are also placed with the program name. Columns 73 through 80 (card sequence no.) are placed with the name for printing with the first header only.
PURG	M	Force the program address counter to M. This instruction must force the address forward by at least one location from the current address.
VAL	A, B, C, . , . ,	For as many values as are present, enter that value into sequential locations.
(* IN COL. 1)		The entire content of this card becomes a remarks card and is placed in the remarks block at the top of the coding from under the following rules: 1. An absolute maximum of 56 cards are allowed. 2. Regardless of their position in the source: a. They are printed at a rate of 7 per page (starting on page 1) until they are all printed or until 56 are printed or the last page is printed where there are less than 8 pages in the assembly.

D VALUES*

0 = END OF RECORD
1 = STATUS bit 0
2 = STATUS bit 1
3 = STATUS bit 2
4 = STATUS bit 3
5 = STATUS bit 4
6 = STATUS bit 5
7 = STATUS bit 6

8 = STATUS bit 7
9 = STATUS bit 8
10 = STATUS bit 9
11 = STATUS bit 10
12 = STATUS bit 11
13 = PARITY ERROR
14 = REJECT
15 = CONTINUE SW. ONLY

* Used to exit from continuous codes.

SMM17 EDIT ROUTINE

(EDTA3E Test No. 3E)

I. INTRODUCTION

The edit routine is a service program for tailoring an SMM17 library, version 3.0 or later. The basic function of EDT is a copy. Copy operations are performed from the current input media position to the end of a target test. Thus, the positions of tests on the input library must be known to build a new library. If the contents of the input library are unknown, a library list may be selected at system load time by setting a bit 0 of the SMM parameter. Only target tests can be modified when writing them to the new library. All other copied tests are copied as they were. Target test modifications that may be selected include prestored equipment address, prestored parameters, and program patches. A copy to a dummy output media affects skipping of unwanted tests on the input library. However, the Skip operation is to the start rather than end of the target test. Special functions, such as rewind and write file mark, may also be selected. All of the above operations are defined by the operator in a function table. The same function table that defines the copy also defines the verify.

II. REQUIREMENTS

A. Hardware

1. A 1700 series computer.

2. One of the following input devices:

   a. Paper Tape Reader

      1721
      1722
      1777

   b. Card Reader

      1726/405
      1728/430
      1729-2
      1729-3

c. Magnetic Tape

1731/601
1732/608-609
1732-3

d. Rotating Mass Storage (if selected as input, cannot be used as output)

1738/853-854
1739
1733
1733-2

3. One of the following output devices:

a. Paper Tape Punch

1723
1724
1777

b. Card Punch

1728/430

c. Magnetic Tape

1731/601
1732/608-609
1732-3

d. Rotating Mass Storage (if selected as output, cannot be used for input)

1738/853-854
1739
1733
1733-2

4. The memory requirement is 8K.

B. SOFTWARE

The program operates under control of the SMM17 monitor.

C. ACCESSORIES

An SMM17 library in relocatable binary (RBD) format is required for input for copy and/or verify operations.

III. OPERATIONAL PROCEDURE

A. LOADING PROCEDURE

The program must be loaded under SMM17 as test number 3E. The equipment address placed in the Q register during build test list must be the equipment address for status 1 of the device which will be used to read the new library during the verify portion of the edit. This equipment address is different than the equipment address used by the write device in the case of paper tape or cards. If no verify is desired, the equipment address must be 0000. The equipment address may be changed after load time by manually entering the new equipment address at the programs initial address plus six (IA+6).

B. PARAMETER STOP

First stop (overflow light on).

(A1) = 3E81 - Test ID word

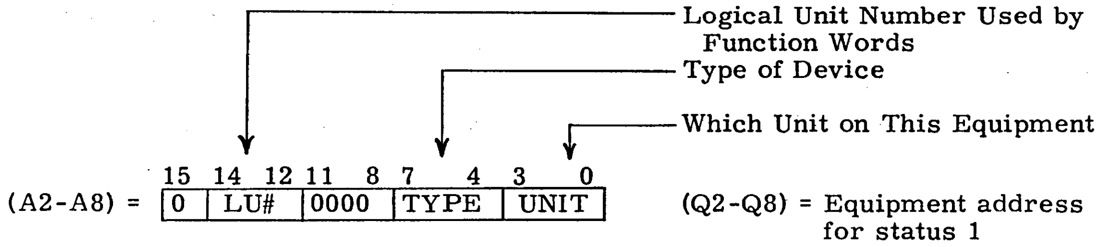
(Q1) = Stop/Jump parameter

Bit 0 - Stop to define logical units
1 - Stop after each function table word pair
2 - Stop after edit complete
3 - Stop on error
4 - Not used
5 - Repeat current function table word pair
6 - Repeat edit
7 - Not used
8 - Omit typeouts
9 - Bias return address display
10 - Not used
11 - Not used
12 - Not used
13 - Not used
14 - Not used
15 - Verify only - do not copy

Stops 2-8 (logical unit definition)

Stops 2 through 8 enable the operator to define logical units (LU) to be used in function table entries. The logical unit concept used here is similar to 1700 MSOS use. Logical unit definition allows a device to be referred to by a 3-bit number rather than by equipment type, equipment address, and unit number. The parameter stops 2 through 8 define logical units 1-7 respectively. The logical unit information must be entered in sequence (1-7). The logical unit definition word pairs use the format as follows:

LOGICAL UNIT DEFINITION WORD PAIR


Logical Unit Number Used by Function Words
Type of Device
Which Unit on This Equipment
15 14 12 11 8 7 4 3 0
(A2-A8) =
0
LU#
0000
TYPE
UNIT
(Q2-Q8) = Equipment address for status 1

Type of Device (same codes as SMM loader types)

0001 = Paper Tape

0010 = Punched Cards

0011 = Magnetic Tape

0100 = Rotating Mass Storage (1738 or 1733)

0101 = Rotating Mass Storage (1739 or 1733-2)

Example 1

Consider a system with a configuration as follows:

Equipment Type	Equipment Number	Unit Number	Channel
1738	3	1	A/Q
1732/608	7	2	1706 #1
1732/609	7	3	1706 #1
1728/430	C	None	A/Q
1721	1	None	A/Q
1723	1	None	A/Q

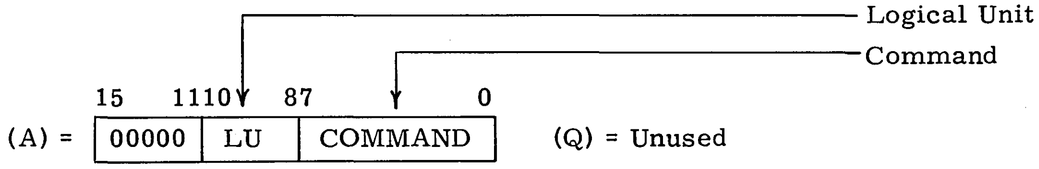
A reasonable logical unit assignment in the parameter stop would be as follows:

A1 = 3E81	Q1 = Stop/Jump parameter		
A2 = 1041	Q2 = 0181	LU#1	1738/853
A3 = 2032	Q3 = 1381	LU#2	1732/608
A4 = 3033	Q4 = 1381	LU#3	1732/609
A5 = 4020	Q5 = 0641	LU#4	1728/430 Punch
A6 = 5020	Q6 = 0621	LU#5	1728/430 Reader
A7 = 6010	Q7 = 00C1	LU#6	1723
A8 = 7010	Q8 = 00A1	LU#7	1721

C. BUILD FUNCTION TABLE

After the last parameter stop "build function table" will be typed and the 1700 will stop for function word entries into the A and Q registers. The operator must enter the function table with some combination of the three types of function word pairs (A and Q registers) to build the desired new library. The three types of function word pairs are as follows: Special function, Skip function, and Copy function. If a verify has been enabled, the function table must begin with Special functions to rewind all the magnetic tapes and disks used in the edit. Following the rewind functions, some combination of Copy functions and Skip functions normally occurs. If the new library is to be on magnetic tape or disk, the last function in the table must be a Special function to write a file mark. The function table is terminated by placing 0000 in both A and Q of a function word pair. A description of the three formats of function word pairs follows:

1. Special Function



The command in bits 0-7 will be executed on the logical unit in bits 8-10. Legal commands (by type of device) are as follows:

Paper Tape (Type 1)

0000 = Feed tape (between target tests)

Punched Cards (Type 2)

0001 = Pick card

0010 = Offset card (430 or 1729-2 only)

0100 = Gate card to secondary hopper (405 only)

Magnetic Tape (Type 3)

0101 = Write end of file (done at end of library)
0110 = Search file forward (1732 only)
0111 = Search file reverse (1732 only)
1000 = Rewind to load point (start of copy)
1100 = Rewind unload

Rotating Mass Storage (Types 4 and 5) Logical Output Unit Only

0000 = Return to zero seek (start of copy)
0001 = Write software end of file (done at end of library)
0010 = Write address headers on track specified by Q
0011 = Write address header from track specified by Q to end of pack

NOTE

Write address headers applicable only on 1738.

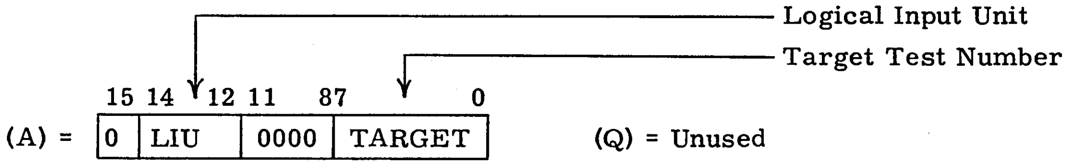
Example 2

Problem: Given the system configuration and logical unit assignments of example 1, rewind both magnetic tapes and disk pack.

Solution:

A	Q	Description
0208	0000	Rewind the 608
0308	0000	Rewind the 609
0100	0000	Return to zero seek on 853

2. Skip Function



The library on the logical input unit in bits 12-14 will be searched to the start of the test in bits 0-7. This function must be followed by a copy function from the same input logical unit.

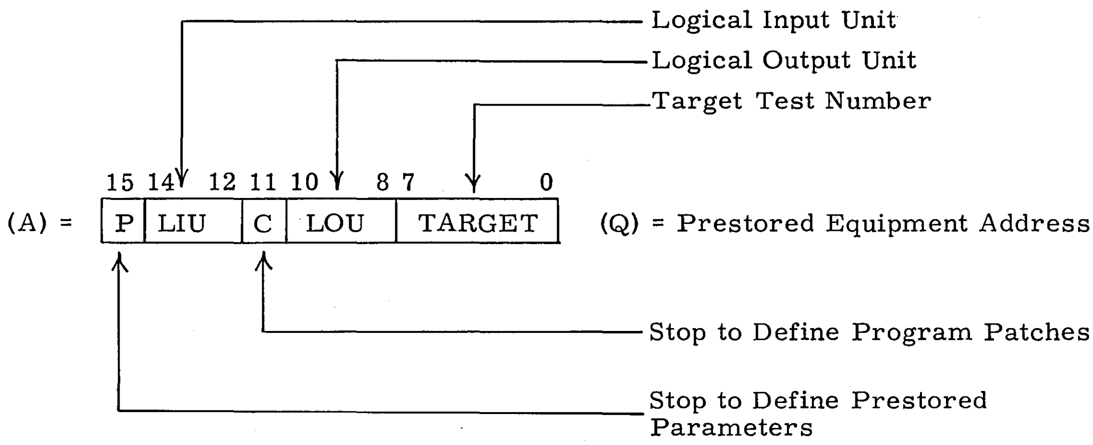
Example 3

Problem: Given the system configuration and logical unit assignments of example 1, search a library on the 608 to test number 15.

Solution:

A	Q	Description
0208	0000	Rewind the 608
2015	0000	Search to test 15

3. Copy Function


Logical Input Unit
Logical Output Unit
Target Test Number
15 14 12 11 10 8 7 0
(A) =
P
LIU
C
LOU
TARGET
(Q) = Prestored Equipment Address
Stop to Define Program Patches
Stop to Define Prestored Parameters

The library on the logical input unit in bits 12-14 will be copied from its current position through the end of the test in bits 0-7 onto the logical output unit in bits 8-10.

Example 4

Problem: Given the system configuration and logical unit assignments of example 1, build a disk pack library with all the tests for the configuration in example 1 from a 7-track magnetic tape library containing Quick Look, SMM, and all SMM tests in numerical order.

Solution:

A	Q	Description
0208	0000	Rewind the 608
0100	0000	Return to zero seek on 853
2105	0000	Copy QL, SMM and test 01-05
2008	0000	Search 608 to test 08
210A	0000	Copy test 08-0A
200D	0000	Search to test 0D
210D	0000	Copy test 0D
2015	0000	Search to test 15
2115	0000	Copy test 15
0101	0000	Write software end of file
0000	0000	Terminate function table

a. Prestored Equipment Address

If the Q register of a function word pair is not equal to zero, the contents of Q will be written onto the logical output unit as the prestored equipment address at the target test's initial address plus six (IA+6). If Quick Look (test number FF) or SMM (test number 00) is the target test, the Q register should be left equal to zero.

Example 5

Problem: Given the system configuration and logical unit assignments of example 1, punch a paper tape with Quick Look, SMM, and tests 08, 0D, and 15 from a disk library containing Quick Look, SMM, and tests 01, 02, 03, 04, 05, 08, 0D, and 15. Prestore the equipment number into tests 08, 0D,and 15.

Solution:

A	Q	Description
1600	0000	Punch Quick Look and SMM
1008	0000	Search to test 08
1608	0181	Punch test 08
160D	0601	Punch test 0D
1615	1381	Punch test 15
0000	0000	Terminate function table

b. Stop to Define Prestored Parameters

If bit 15 of A in a function word is a one, a simulated parameter stop for the test in bits 0-7 will occur when the logical input unit reaches the relocatable binary data block (RBD) in which that test's parameters reside. The parameters selected in the parameter stop will be written onto the logical output unit as the target test's new prestored parameters. In the simulated test parameter stop, the Stop/Jump parameter is replaced with the new prestored equipment address. The equipment address cannot be changed at this time. SMM prestored parameters are a special case and, therefore, must be handled differently. To change the prestored SMM parameters, the target test must be Quick Look (test number FF). In the simulated SMM parameter stop, the Stop/Jump parameter is displayed and may be altered. If SMM (test number 00) is the target test in a function word that requests a stop to define parameters, the operator is allowed to define SMM's prestored test list including the optional entries of equipment address, test load address, and Stop/Jump parameter. In the first series of stops, the operator must enter the test and frequency into A and may enter the equipment address into Q. A maximum of ten pairs of entries may be made. The list must be terminated by placing zeros in both A and Q. In the second series of stops, the operator may enter the test load address in A and the Stop/Jump parameter in Q. The number of stops in the second series will equal the number of entries made in the first series of stops.

Example 6

Problem: Given the system configuration and logical unit assignments of example 1, punch a card deck with Quick Look, SMM, and test 08 from a disk library containing Quick Look, SMM, and tests 01, 02, 03, 04, 05, 08, 0D, and 15. In test 08, prestore the equipment address and change the prestored parameters to run unit 1 instead of unit 0.

Solution:	A	Q	Description
Function			
Table	1400	0000	Punch Quick Look and SMM
Word	1008	0000	Search to test 08
Pairs	9408	0181	Punch test 08
	0000	0000	Terminate function table

After reading the RBD block in which the parameter for test 08 resides, a simulated test 08 parameter stop will occur as follows:

	A	Q	Description
Simulated	0831	0181	ID Word/Equipment address
Parameter	119F	0063	Sections-unit/cylinders
Stops	0004	0000	Interrupt line/not used

The operator would change A of the second stop from 119F to 319F. This would select unit 1 to be run instead of unit 0.

c. Stop to Define Program Patches

If bit 11 of the function word in A is a one, stops will occur when the logical input unit reaches the test in bits 0-7 to allow program patches to be entered. These program patch stops will occur at two different positions within the target test. The first stop will occur after the first block of the target test has been read. At this stop, A will contain the current function word and Q will be zero. The operator must enter into Q the number of words to be added to the length of the test. If all patches are to be made to code already within the test's boundaries, Q should remain zero. After the length change stop, the target test will be copied to the logical output unit until the last block of the target test (or overlay) has been read. Additional stops are then made to allow the actual program patches to be entered. The first such stop will occur with A containing the current function word and Q containing zero (or the number of the overlay being changed). The operator must enter the listing address of the program patch into A and the data into Q. If the data in Q is an address that must be biased at load time, bit 15 of A must be set to a one. The program patch stop must be repeated for each patch. When all patches have been made to the test (or overlay), the operator must zero A and Q to terminate patching of the test (or overlay). If the target test contains overlays, an additional stop will occur after A and Q have been cleared. This additional stop allows the operator to bypass all the remaining overlays. When the stop occurs, A contains the current function and Q contains the number of the next overlay to be changed. If the operator does not alter A or Q, the program patch stops (excluding the length change stop) will be repeated for next overlay. If the operator zeros the A register, program patch stops will be bypassed for the remaining overlays.

Example 7

Problem: Given the system configuration and logical unit assignments of example 1, copy to a 9-track magnetic tape library all the tests from a disk library containing Quick Look, SMM, and tests 01, 02, 03, 04, 05, 08, 0D, and 15. In test 08, patch listing address 05DC to contain an NOP instruction.

Solution:	A	Q	Description
Function			
Table	0308	0000	Rewind the 609
Word	1B08	0000	Copy to 08, patch 08
Pairs	1315	0000	Copy remainder of disk
	0105	0000	Write end of file
	0000	0000	Terminate function table

When the first record (name block) of test 08 has been read, a length change stop will occur as follows:

	A	Q	Description
Length Change Stop	1B08	0000	Active function/0000

Since the program patch would not change the length of test 08, the operator would leave Q unaltered and place the RUN/STEP switch into RUN.

When the last record (transfer block) of test 08 has been read, a program patch stop will occur as follows:

	A	Q	Description
Program Patch Stop	1B08	0000	Active function/0000

To enter the program patch the operator would enter 05DC into the A register and 0B00 (an NOP instruction) into the Q register. On the next stop, the operator would clear both A and Q to terminate the program patch stops.

d. Verify

A verify must only be enabled when a library is to be built on a single logical unit. This is because there are provisions for only one verify device equipment address (entered into Q during build test list). After the copy portion of the edit has completed, a message will be typed as follows: HIT RUN TO VERIFY. The computer will stop with 3E10 in A and the Stop/Jump parameter in Q. The Stop/Jump parameter may not be changed at this time. At this stop, the operator must make all paper tapes and card decks ready for the verify. All magnetic tapes and disks will be readied by the program through the rewind special functions at the start of the function table. When all devices have been readied, the operator must place the RUN/STEP switch to RUN. The program then again performs the operations designated in the function table. The only difference between this pass and the copy pass through the function table is that every Write operation is replaced by a read from the verify device and a compare. All prestored paramater entries and program patch entries must be repeated during the verify. Setting bit 15 of the Stop/Jump parameter at the start of an edit causes the copy portion of the edit to be eliminated and only the verify portion is executed. If binary Quick Look has been output to cards or paper tape, the binary Quick Look must be manually bypassed for the verify.

Example 8

Problem: Given the system configuration and logical unit assignments of example 1, build a disk pack library with all the tests for the configuration in example 1 from a 7-track magnetic tape library containing Quick Look, SMM, and all SMM tests in numerical order. Verify the disk pack library.

Solution:

A	Q	Description
0208	0000	Rewind the 608
0100	0000	Return to zero seek on 853
2105	0000	Copy/verify QL, SMM,and test 01-05
2008	0000	Search 608 to test 08
210A	0000	Copy/verify test 08-0A
200D	0000	Search 608 to test 0D
210D	0000	Copy/verify test 0D
2015	0000	Search 608 to test 15
2115	0000	Copy/verify test 15
0101	0000	Write software EOF (on copy only)
0000	0000	Terminate function table

The above function table would be executed once for a copy and the message HIT RUN TO VERIFY would be typed. The computer would stop with 3E10 in A and the Stop/Jump parameter in Q. The operator would place the RUN/STEP switch to RUN. The above function table would again be executed, but for a verify on this second pass.

E. SEQUENCE OF EDIT EVENTS

Operator action during edit will be required in the sequence as follows:

1. Parameter Entry
2. Build Function Table
3. Program Length Stop (if stop to enter program patches has been selected in the function word)

4. Prestored Parameter Stop (if stop to enter prestored parameters has been selected in the function word)

5. Program Patch Stop (if stop to enter program patches has been selected in the function word)

6. End of Function Word Stop (if stop at end of section has been selected in the Stop/Jump parameter word)

7. Hit Run to Verify Stop (if verify has been selected during build test list)

8. End of Test Stop (if stop at end of test has been selected in the Stop/Jump parameter word)

USER CAUTION

If necessary to abort edit, program should be restarted at IA and 0 function entered to exit properly. Otherwise, some SMM control words will be incorrect.

IV. OPERATOR COMMUNICATION

A. MESSAGE FORMATS

1. Normal Program Identification Typeout

During initialization, the program identification typeout occurs as follows:

```
EDTA3E

IA = xxxx
```

2. Normal Build Function Table Typeout

After parameter entry, a typeout occurs as follows:

```
BUILD FUNCTION TABLE
```

The typeout is followed by a stop with the contents of A and Q indeterminate. At this stop the operator must enter a function word pair into the A and Q registers and place the RUN/STEP switch in RUN. This procedure must be repeated until all the desired function word pairs have been entered. Then, to terminate the function table, the operator must clear both A and Q and place the RUN/STEP switch in RUN.

3. Normal Verify Typeout

Before verification, a typeout occurs as follows:

HIT RUN TO VERIFY

The typeout is followed by a stop with the register containing the following:

A = 3E10 Q = Stop/Jump parameter

The Stop/Jump parameter may not be changed at this time. At this stop the operator must make all paper tapes and card decks ready for the verify portion of the edit. To continue, the operator must place the RUN/STEP switch in RUN.

4. Normal End of Function Stop

This stop will occur at the end of each function, if bit 1 of the Stop/Jump parameter is set. The stop format is as follows:

A1 = 3E22	Q1 = Stop/Jump parameter
A2 = Function just completed	Q2 = Not used

To continue, the operator must place the RUN/STEP switch in RUN.

5. Normal End of Test Stop

This stop will occur at the end of the copy (and/or verify if selected), if bit 2 of the Stop/Jump parameter is set. The stop format is as follows:

A1 = 3E24	Q1 = Stop/Jump parameter
A2 = Pass count	Q2 = Not used

If more than one pass of EDT has been selected, the operator may continue by placing the RUN/STEP switch in RUN.

6. Error Stop

An error stop will occur only if bit 3 of the Stop/Jump parameter is set. The basic stop format is as follows:

A1 = 3 E x 8	Q1 = Stop/Jump parameter
A2 = Error code	Q2 = Return address

The error codes are described in the Message Dictionary. The content of A and Q will be given for only the third and subsequent stops.

V. MESSAGE DICTIONARY

Code	Description
02	Loader encountered illegal RBD block type. Edit restarted if no load error.

A3 = Block type
A4 = Current function word

Q3 = Last test number
Q4 = Not used

03 No RBD transfer block found. Edit restarted if no load error. This error may occur as a result of not following a Skip function with a Copy function from the same logical unit, since a skip terminates after read a NAM block of the Skip function target test.

A3 = Faulty test number
A4 = Current function word

Q3 = Last test number
Q4 = Not used

13 Checksum error (cards or paper tape). Attempt reload.

A3 = Not used
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

14 Paper tape reader alarm. Attempt reload.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

15 Card reader alarm. Attempt reload.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

16 Card reader generated end of operation on column other than column 80. Try reload.

A3 = Column count
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

Code Description

17 Input unit magnetic tape parity error. Load aborted on eighth retry, successful if less.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Retry count
Q4 = Data address if block type 4050
Q5 = Current function word

18 Disk pack alarm. Load aborted on 50th retry, successful if less.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Retry count
Q4 = Data address if block type 4050
Q5 = Current function word

20 Paper tape punch alarm. Driver will punch a block of equal length and re-punch record when punch ready. Error in an RBD block (4050) is recoverable if past the eighth frame of the block. Error in a NAM block (2050) or an OVL block (E050) may cause load problems.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

21 Paper tape supply low. Punching continues to gap between tests.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

22 Punch alarm on 430. Card is offset and repunched.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Column count if punch error (stop A3 bit 8) status 2 if not punch error
Q4 = Data address if block type 4050
Q5 = Current function word

23 Output unit magnetic tape alarm.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Not used
Q4 = Data address if block type 4050
Q5 = Current function word

Code Description

24 Output unit magnetic tape alarm. Record gap lengthened over bad spot on eighth retry.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Block write retry count
Q4 = Data address if block type 4050
Q5 = Current function word

25 Disk pack alarm on output or special function.

A3 = Status
A4 = Block type
A5 = Last test number

Q3 = Disk address (except function rewind)
Q4 = Data address if block type 4050
Q5 = Current function word

30 Verify data compare error.

A3 = Verify unit data
A4 = Block type
A5 = Last test number
A6 = Failing word number (words numbered from 1)

Q3 = Expected data
Q4 = Data address if block type 4050
Q5 = Current function word
Q6 = Number of words being verified in this block

60 Non-standard 6000 class assembler NAM block encountered. Clear A and Q of the second stop and enter the date and test biasing code in the form as follows:

A = mmdd

Q = yy0x

where mm = month (1-9, 10-12 hex)
dd = day (1-9, 10-19, 20-29, 30-31)
yy = year (70-79 hex)
x = 0 for monitor biasing or 1 for test biasing

No typeout occurs for this error.

VI. GLOSSARY

LOGICAL UNIT: A system-level software concept (as opposed to the I/O driver level, physical unit). It allows tabulated physical information words (equipment address, device type i.e. mag. tape, etc.) necessary to function and operate a specific physical unit to be referenced by the relative position, or index of that particular set of physical information words within the system's table of physical unit parameters, such table defining those units available to the system at a given installation. For example, logical unit no. 1 refers to the first set of physical parameters in the table, logical unit no. 2 to the second set, etc.

LIU/LOU: Abbreviations for logical input unit and logical output unit. The logical unit number indexes the physical unit table built in EDT's parameter entry routine and is entered in the upper Hex-digit of edit's function word for the desired input unit (LIU) of the operation described by that function word, or in the next-to-upper Hex-digit for the output unit (LOU) of that operation.

FUNCTION WORD: One 16-bit word entered by the operator, via the A register, to edit's function table. It describes a single edit operation and the logical unit(s) involved. The table, consisting of at least one, but usually more, function words, is executed one word at a time to perform the desired, operator defined, SMM17 library editing operations. Possible operations are:

1. Skip (LOU = zero; expects to be followed by a copy from the same LIU)
2. Copy (neither LIU nor LOU = zero)
3. Special function (LIU = zero) i.e., rewind library on mag. tape or disk pack, write end-of-library (file mark) on mag. tape or disk pack, punch leader on paper tape, etc.

TARGET TEST: An 8-bit hexadecimal test identification number (i.e. COM=$01, MEM=$02, etc.) placed in the lower byte of edit's function word for a Skip or Copy operation. It is used by edit to flag the end of that Skip or Copy operation and the subsequent read-up and execution of the next function word in the function table. Any program containing overlays must be designated as a target test when editing the "raw" output from the assembler. Edit adds overlay blocks.

SPECIAL FUNCTION: If the LIU of the function word is zero, there is no input unit defined for that operation; hence, it is neither a Skip nor a Copy operation. The lower 8-bits of the function word, in this case, will determine the actual operation, according to the output device type. For example, if the LOU points to physical information for a disk pack, and the lower 8-bits = $01, a special data sector is written which will be decoded when read by the DP input routine as "end-of-file" and result in rewinding the library, i.e. positioning the DP, to the beginning.

3000 CHANNEL SIMULATOR PROGRAM UPDATE

(UD302D Test No. 2D)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Runs on an SC/1700 in the following configuration:
   a. (2) Tape Units
   b. (1) Teletype
   c. (1) 1729 Card Reader (Optional) *
   d. A minimum of 8K of core
   e. The number of edit entries are limited by:
      1) Core size. A maximum of 9600 characters (including spaces) can be entered into 8K, and an additional 8182 characters for each additional 4K of core. Trailing spaces not included.
      2) Or a maximum of 1000 entries, whichever comes first.

2. UD3 overlays SMM17.

B. LOADING PROCEDURE

1. Call as standard SMM17 test No. 2D

C. PARAMETERS

1. If the SKIP switch is set, typeout is suppressed.

II. CALLS

A. EDIT (.)

1. .DAAAA,BBBB
   Delete AAAA0 (If only entry) or AAA0 through BBBB0.

2. .EAAAA
   End update after transferring card AAAA0 to Unit 1, an EOF mark is then written to Unit 1.
   Tape handling begins after this entry.

3. .FAAAA,BBBB
   Fetch card AAAA0 (if only entry) or AAAA0 through BBBB0. These cards, from Unit 2, are typed and given an entry number. They are now data entries as if entered via TTY.

* If the card reader is available, has power on, and is loaded with cards, the cards will be read and handled as if they were teletype entries. The card columns must be correct. Control is returned to the TTY when the card reader faults depress rub-out to continue.

4. .IAAAA

   Insert the following entries after card AAAA0.

5. .KAAA,BBB

   Correct previous entry AAA (if only number entered) or AAA through BBB. After the corrections are complete, entry numbers jump back to normal sequencing.

6. - (minus sign)

   A minus sign will delete an edit entry, of which the entry number is showing. This entry can be used in conjunction with the .K option to strike a bad edit entry.

7. .T

   Terminate EDIT and begin tape handling. An EOF mark is written to Unit 1 after the last EDIT entry.

B. CONTROL (/)

1. /UX (X = Units 0 through 2 only)

   Unload unit X.

2. /MX

   Mark (Write an EOF on) unit X.

3. /SFFX

   Search file forward on unit X.

4. /SFBX

   Search file mark backward unit X. (1732 only)

5. /RX

   Rewind unit X.

III. OPERATION

A. REQUIRED ENTRIES

1. In response to message:

   "MTO = ICE, MT1 = ICE, MT2 = ICE"

   Enter the interrupt line, converter, and equipment numbers for units 0, 1, and 2 respectively.

2. In response to message:

   "ID = "

   Enter the three characters desired in columns 73 through 75 of the sequence number on the output (Unit 1).

NOTE

New source is automatically sequence numbered by tens.

3. Enter any of the commands in II needed to perform the update. Any entry, other than a command, is considered to be a data entry and the following rules apply.

   a. If the entries are from TTY, tabs are generated when a space is entered. Cards are required to carry the proper tabs but column skipping is done when spaces are encountered to reduce false data entries.

   b. Auto tabbing is suppressed when:

      1) An * is in column 1.
      2) After column 20 on a BCD entry.
      3) After column 41.

   c. The tabs (columns) for this program are 10, 20, and 41.

4. Tape assignments are:
   Unit 0 = Old Source
   Unit 1 = New Source
   Unit 2 = Fetch (Card Images)

5. With the exception of Unit 2, no tape operations are performed as the result of an EDIT entry before an.E or .T entry is made.

6. When the EDIT is complete, the TTY will output Message III. A. 2. and the program is ready to start the next job.

1700 SOURCE/6000 TVC UPDATE

(UD1A57 Test No. 57)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Does not run with SMM17 (uses for loading only).
2. Requires two tape units and one teletype.
3. Requires the 1700 source to be at load point and end with an EOF mark.
4. Requires the 6000 external listing tape from a "TVC" assembly to be at load point on its tape. (Stacked programs or blocked tape are not acceptable.)
5. Requires special control cards in the 1700 source prior to a run with this program. (See Figure 1, types 5 and 6.)
6. Updates only one program at a time.
7. Sets P=2 and runs to reload SMM17.

B. LOADING PROCEDURES

1. Standard SMM17 call.
2. Test number 57.

C. PARAMETERS

1. Entry via teletype. (See II B.)

D. OPERATING INSTRUCTIONS

1. Load UPD via SMM17 operation instructions.
2. Mount a scratch tape on TU1.
3. Mount 1700 source tape on TU0.
4. Respond to teletype requests. (See II B.)
5. When TU0 rewinds, unload (if using a 2-tape system) and mount the 6000 listing tape on TU0. Change TU0 to TU2.
6. Depress and release the SKIP switch.
7. When TU2 rewinds, unload (if using a 2-tape system) and remount the original 1700 Source Tape. Change TU2 to TU0.
8. Depress and release the SKIP switch.
9. When the teletype goes back to request entries again, the job is complete and TU1 is holding the new source tape.

II. MESSAGES

A. NORMAL MESSAGES

1. Start UD1A57 IA=0000

B. COMMAND MESSAGES

1.
```
MTO=ICE, MT1=ICE, MT2=ICE
    XXX       YYY       ZZZ
```

   This message is requesting the interrupt line, converter number, and equipment number respectively for tape units (TU)0, 1, and 2 respectively.

2. BC NAME = AAAAAAAA

   This message is requesting a name of up to eight characters (less with a carriage return) which exactly matches the name in the 1700 source special control cards. (See Figure 1, types 5 and 6)

C. ERROR MESSAGES

1. None

	01	03	07	11	13	17	18	26	30	35	41	72
1	*	REMARKS CARDS										
2		NUM	$HHHH		AAA	SSSSSSSS		OOO	RRRR	RRRRR	COMMENTS	
				NLST								
3		NUM	$0000									
		NUM	$10000									
		NUM	$00000									
				LST								
4				EJT								
	*			NNNNNNNN					PAGE	NNNN		
5*	*			STRT			NNNNNNNN					
6*	*			END			NNNNNNNN					

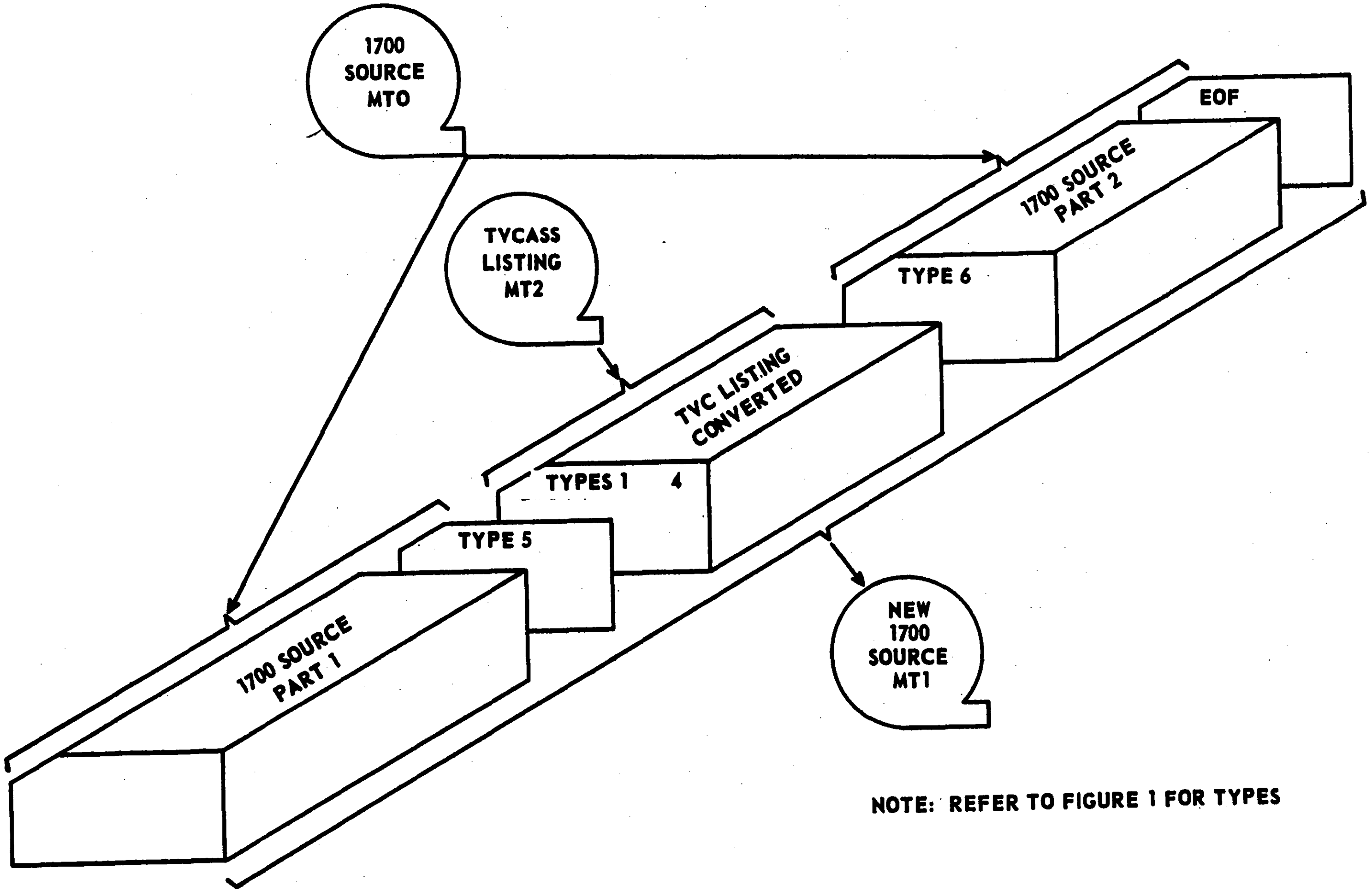
*These types are not made by the Edit program (in 1700 source).

The above shows the types of cards generated by the BC2 Edit program.

The program is extracted from a 6000 External Listing tape of the TVC assembly and inserted between type 5 and 6 cards bearing the same name as on those cards and specified by teletype. Up to an 8-character name is permitted. Types 5 and 6 are 1700 rem cards and inserted in the 1700 source some time prior to the use of BC2 Edit. The following is a list of the types generated for the respective TVC code:

TVC	GENERATES TYPE
Page eject	4
Rem card	1
BSS card/ORG card	1 for the statement, 3 with as many NUM $0000 as necessary to fill the block requirement.
INST. card	2
END card	1
TIME and all other non instruction types	
CON/code types	2

Figure 1.



NOTE: REFER TO FIGURE 1 FOR TYPES

Figure 2

1700 SOURCE/6000 BUCAL UPDATE

(UD2A58 Test No. 58)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Does not run with SMM17 (uses for loading only).
2. Requires two tape units and one teletype.
3. Requires the 1700 source to be at load point and end with an EOF mark.
4. Requires special control cards in the 1700 source prior to a run with this program. (See Figure 1, types 5 and 6.)
5. Requires that the 6000 external listing tape from a BUCAL assembly carry the same name in its identification card in the 1700 source. (See Figure 1, types 5 and 6.)
6. Updates only one program at a time.
7. Sets P=2 and runs to reload SMM17.

B. LOADING PROCEDURES

1. Standard SMM17 call.
2. Test number 58.

C. PARAMETERS

1. Entry via teletype. (See II B.)

D. OPERATING INSTRUCTIONS

1. Load UPD via SMM17 operation instructions.
2. In response to message II B 3, mount a scratch tape on TU1 and 1700 source tape on TU0.
3. Respond to teletype requests. (See II B.)
4. In response to message II B 4, unload (if using a 2-tape system) and mount the 6000 listing tape on TU0. Change TU0 to TU2.
5. Depress and release the SKIP switch.
6. In response to message II B 5, unload (if using a 2 tape system) and mount the original 1700 source tape on TU2. Change TU2 to TU0.
7. Toggle SKIP switch.

8. When the teletype goes back to request entries again, the job is complete and TU1 is holding the new source tape.

II. MESSAGES

A. NORMAL MESSAGES

1. Start UD2A58 IA=0000

B. COMMAND MESSAGES

1.
```
MTO=ICE, MT1=ICE, MT2=ICE
     XXX       YYY       ZZZ
```

   This message is requesting the interrupt line, converter number, and equipment number respectively for tape units (TU)0, 1, and 2 respectively.

2. BC NAME = AAAAAAAA

   This message is requesting a name of up to eight characters (less with a carriage return) which exactly matches the name in the 1700 source special control cards and the identification card for the Bucal Assembly. (See Figure 1, types 5 and 6)

3. MTO=BCX, MT1=SCRATCH (self explanatory).

4. Load BUCAL list tape on MT2 and toggle SKIP switch (self explanatory).

5. Reload original BCX source on MTO and toggle SKIP switch (self explanatory).

	01	03	07	11 13	17	18	26	30 35	41	72
1	*	REMARKS		CARDS						
2		NUM	$HHHH	AAA	S	SSSSSSS	OOO	RRRRRRRRR	COMMENTS	
				NLST						
3				ORG		PRGBUF +	$NNNN			
				LST						
4				EJT						
5*	*			STRT		NNNNNNNN				
6*	*			END		NNNNNNNN				

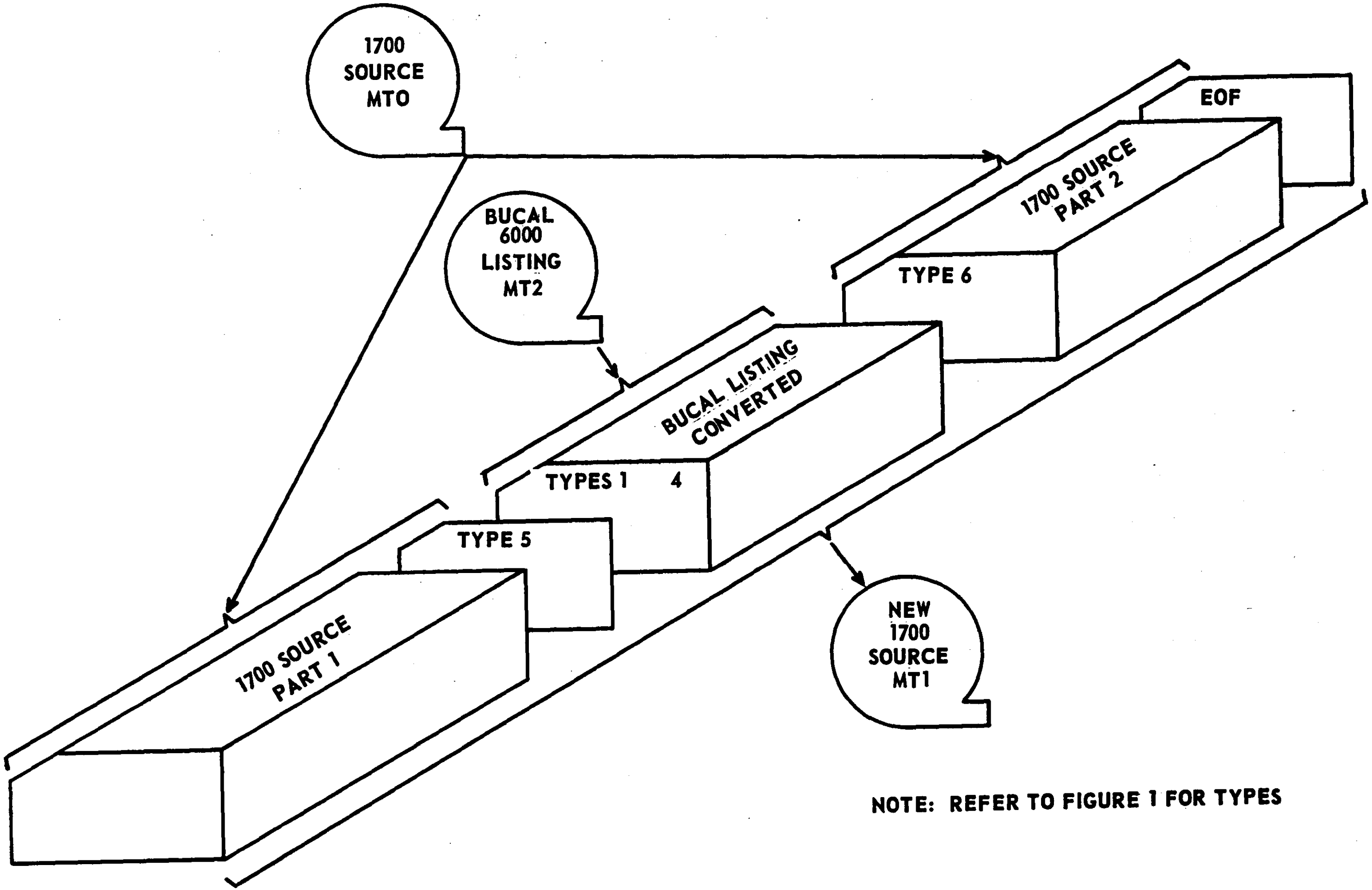
*These types are not made by the Edit program (in 1700 source).

The above shows the types of cards generated by the BC2 Edit program.

The program is extracted from a 6000 External Listing tape of the BUCAL assy. and inserted between type 5 and 6 cards bearing the same number as on those cards and specified by teletype. Up to an 8 character name is permitted. Types 5 and 6 are 1700 rem cards and inserted in the 1700 source some time prior to the use of BC2 Edit. The following is a list of the types generated for the respective BUCAL code:

BUCAL	GENERATES TYPE
Page eject	4
Rem card	1
BSS card /ORG card	1 for the statement and 3
INST. card	2
END card	1
TIME and all other non instruction types	
CON/code types	2

Figure 1



NOTE: REFER TO FIGURE 1 FOR TYPES

Figure 2

1700 COMMAND TEST
(COM001 Test No. 1)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Tests of instructions that alter the Mask register will return control after each pass of the section.
2. In case only a 4-bit Mask register is present, the contents of M will be changed accordingly.
3. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. The test operates as a subprogram under control of the 1700 System Maintenance Monitor (SMM17).
2. The calling sequence is that specified by SMM17.
3. The test can be restarted by MC, set P = IA and RUN.

C. PARAMETERS

1. Normal operation requires no parameters.
2. To alter parameters, use the procedures explained in SMM17. The identification word and Stop Jump word are then displayed on a parameter stop. Next parameters are tests to be run (A) and the numbers of sets of operands to be used (Q).
3. Display Format:

A =

15 8	7 0
First test section to be tested	Last test section to be tested

All tests between the first and last section numbers are tested. If one test section is desired, the sections entered are the same. The maximum range is 1_{16} - 43_{16}.

Q =

15 0
Number of sets of operands used

Any positive number can be used.

4. SELECTIVE JUMP and SELECTIVE STOP key setting

   a. SELECTIVE STOP must be set.

   b. Use SELECTIVE SKIP switch as outlined in the SMM17 Reference Manual.

D. MESSAGES

1. Typeouts or Alarms

   a. Normal Program Typeouts

      1) Command test identification at start of test
         COM001 COMMAND TEST
         IA = XXXX, FC = XX CP2F, VER. 3.1

      2) End of Command test

A	Q	A	Q
0104	STOP	PASS NO.	RETURN ADDRESS

   b. Error Alarms

      1) The following is typed out:

         a) Identification word

         b) Stop/Jump parameter

         c) Section and error number

         d) Return address

         e) Correct data or simulated results

         f) Incorrect data

      2) See each individual section write up for its specific error alarm.

   c. Error Codes - See individual section write up.

E. ERROR STOPS

1. The same data typeout for error display will be displayed in A and Q.

2. See each individual section write in for its specific error.

II. DESCRIPTION

A. PROGRAM DESCRIPTION

This test checks internal 1700 instructions with fixed and random operands. The instructions are tested beginning with the simpler ones and ending with the more complex.

The check of actual results is made by either a simulation or a comparison to anticipated results. The writeups of the individual tests follow and are in the same order as they appear in the program.

Section 1 JUMP

Error Number 0

Test if Jump instruction is executed and if destination is correct. If destination is ±1 instruction, a Stop should occur.

Error Display: Standard error format not used.
A = 108 and OVERFLOW light will not be lit.

Section 2 RETURN JUMP

Error Number 1
Incorrect address was stored in return address.
Error Display: A = correct address, Q = incorrect address

Section 3 LOAD A

Error Number 1
An operand was loaded into A. The same operand and the contents of A are then compared via an EXCLUSIVE OR.

Error Display: A = correct data, Q = incorrect data

Section 4 STORE A

Error Number 1
An operand was loaded into A and stored. A was then compared with the operand stored via an EXCLUSIVE OR.

Section 5 LOAD Q

Error Number 1
An operand was loaded into Q and transferred to A. The same operand and the contents of Q were then compared via EXCLUSIVE OR.

Error Display: A = correct data, Q = incorrect data

Section 6 STORE Q

Error Number 1
An operand was loaded into Q and A. Q was then stored and the stored operand was compared to (A).

Error Display: A = correct data, Q = incorrect data

Section 7 SKIP IF A = +0

Error Number 1
No Skip occurred when positive zero was loaded into A and then tested.
Error Display: No data display

Error Number 2
A Skip occurred when negative zero was tested in A.
Error Display: No data display

Error Number 3
Positive zero Skip did not occur for contents of A, but when tested in Q, a Skip occurred.

Error Display: A = contents of A, Q = contents of Q

Error Number 4
Positive zero Skip occurred when tested in A, but did not occur when tested in Q.
Error Display: A = contents of A, Q = contents of Q

Section 8 SKIP IF A ≠ 0

Error Number 1
No Skip occurred when negative zero was loaded into A and tested.
Error Display: No data display

Error Number 2
A Skip occurred when zero was loaded into A and tested.
Error Display: No data display

Error Number 3
A was tested with SAN instruction and no Skip took place, but when tested by SAZ instruction it was found to be non-zero.
Error Display: A = contents of A

Error Number 4
A was tested with SAN instruction and a Skip took place, but when tested by SAZ instruction it was found to be zero.
Error Display: A = contents of A

Section 9 SKIP IF A = + (positive)

Error Number 1
When A was loaded with an operand and tested for positive value a Skip did not occur. When complemented and tested again the Skip did not occur.

Error Display: A = operand tested first, Q = complement of operand

Error Number 2
When A was loaded with an operand and tested for positive value, a Skip occurred. When complemented and tested again, the Skip occurred.

Error Display: A = operand tested first, Q = complement of operand

Section A SKIP IF A = - (negative)

Error Number 1
When A was loaded with an operand and tested for negative value, a Skip did not occur. Then A was tested for positive and found to be negative.

Error Display: A = operand tested

Error Number 2
When A was loaded with an operand and tested for negative value, a Skip occurred. Then A was tested for positive and found to be positive.

Error Display: A = operand tested

Section B SKIP IF Q = +0

Error Number 1
Q and A were loaded with the same operand, Q was then tested with SQZ, and no Skip occurred. But when A was tested with SAZ, a Skip occurred.

Error Display: A = operand tested

Error Number 2

Q and A were loaded with the same operand, Q was then tested with SQZ, and a Skip occurred. But when A was tested with SAZ, a Skip did not occur.

Error Display: A = operand tested

Section C SKIP IF Q + 0

Error Number 1

Q and A were loaded with the same operand; Q was then tested with SQN and no Skip occurred. But when A was tested with SAN, a Skip occurred.

Error Display: A = Operand tested

Error Number 2

Q and A were loaded with the same operand, Q was then tested with SQN and a Skip occurred. But when A was tested with SAN, a Skip did not occur.

Error Display: A = operand tested

Section D SKIP IF Q = + (positive)

Error Number 1

Q and A were loaded with the same operand, Q was then tested with SQP, and no Skip occurred. But when A was tested with SAP, a Skip occurred.

Error Display: A = operand tested

Error Number 2

Q and A were loaded with the same operand, Q was then tested with SQP, and a Skip occurred. But when A was tested with SAP, a Skip did not occur.

Error Display: A = operand tested

Section E SKIP IF Q = - (negative)

Error Number 1

Q and A were loaded with the same operand, Q was then tested with SQM, and no Skip occurred. But when A was tested with SAM, a Skip occurred.

Error Display: A = operand tested

Error Number 2

Q and A were loaded with the same operand; Q was then tested with SQM and a Skip occurred. But when A was tested with SAM, a Skip did not occur.

Error Display: A = operand tested

Section F A LEFT SHIFT AND TEST FOR OVERFLOW

An operand was loaded into A and shifted left a predetermined number of times. A check is made for overflow after shifting.

Error Number 1
The end results of the shifting did not equal the starting value.
Error Display: A = correct data, Q = incorrect data

Error Number 2
Overflow occurred.
Error Display: A = operand, Q = not applicable

Section 10 Q LEFT SHIFT AND TEST FOR OVERFLOW

Error Number and displays are the same as Section F.

Section 11 A RIGHT SHIFT

Error Number 1
A was loaded with 4000, then shifted to the right once and compared to the known value. This is repeated 15 times for each iteration of the test section.

Error Display: A = correct data, Q = incorrect data

Section 12 A RIGHT SHIFT

Error Number 1
Same as Section 11 with the exception that A = 800 instead of 4000 when starting the shifting.

Section 13 A RIGHT, Q RIGHT

Error Number 1
A and Q were loaded with the same operand, then shifted and compared.
Error Display: A = content of shifted A, Q = content of shifted Q

Section 14 LONG LEFT SHIFT AND CHECK FOR OVERFLOW

A and Q were loaded with operands and shifted 1, 2, . . . 31 places then compared to the original value loaded into A and Q and a check is made for overflow.

Error Number 1
Shifted results did not equal original values in A and Q.
Error Display: A = original contents of A, Q = original contents of Q
A = shifted contents of A, Q = shifted contents of Q

Error Number 2
Overflow occurred
Error Display: Inapplicable.

Section 15 LONG RIGHT SHIFT

Error Number 1
A and Q were loaded with 8000 and 4000, then shifted one place and compared to the known result. The shifting continues until the bits have traversed the registers.

Error Display: A = correct contents of A, Q = correct contents of Q
A = incorrect contents of A, Q = incorrect contents of Q

Error Number 2

A and Q were loaded with 8000, then shifted one place and compared to the known results. The shifting continues until the bits have traversed the registers.

Error Display: Same as Error Number 1

Section 16 ENTER A

Error Number 1
A known quantity is masked into an ENA instruction; then the instruction is executed and compared to the known quantity.

Error Display: A = correct data, Q = incorrect data

Section 17 ENTER Q

Error Number 1
The same as Section 16 with the exception that Q is used instead of A.

Section 18 AND WITH A

Error Number 1

A is set to all one's; then an AND is executed with a known operand. The contents of A and the operand are then compared.

Error Display: A = correct data, Q = incorrect data

Section 19 ADD TO A

Error Number 1

The ADD instruction was simulated by shifting and bit comparison. The simulated results were then compared to the instruction results.

Error Display: A = simulated results, Q = instruction results

Section 1A EXCLUSIVE OR WITH A

Error Number 1

The EOR instruction was simulated by shifting and bit comparison. Then the simulated results were compared to the instruction results.

Error Display: A = simulated results, Q = instruction results

Section 1B ADD TO Q

Error Number 1

An ADD instruction was used to simulate the ADQ. Then the ADQ instruction is executed and compared to the simulated results.

Error Display: A = simulated results, Q = instruction results

Section 1C INCREASE A

Error Number 1

An ADD instruction was used to simulate the INA instruction. Then the INA instruction was executed and compared to the simulated results.

Error Display: A = simulated results, Q = instruction results

Section 1D INCREASE Q

Error Number 1

An INA instruction was used to simulate the INQ instruction. Then the INQ instruction was executed and compared to the simulated results.

Error Display: A = simulated results, Q = instruction results

Section 1E REPLACE ADD ONE IN STORAGE AND CHECK FOR OVERFLOW

Error Number 1

An ADD instruction was used to simulate the RAO instruction. Then RAO instruction was executed and compared to the simulated results.

Error Display: A = simulated results, Q = instruction results

Error Number 2

Overflow occurred

Error Display: A = simulated results, Q = instruction results

Section 1F SUBTRACT

Error Number 1

An ADD instruction with a complemented operand is used to simulate the SUB instruction. Then the SUB instruction is executed and compared to the simulated results.

Error Display: A = simulated results, Q = instruction results

Section 20 SKIP ON OVERFLOW, SKIP ON NO OVERFLOW

Error Number 1

An overflow was forced (+ to -), but when tested with SOV the Skip did not occur.

Error Display: No data display

Error Number 2

The execution of a SOV instruction failed to clear an overflow.

Error Display: No data display

Error Number 3

An overflow was forced (- to +), but when tested with SOV the Skip did not occur.

Error Display: No data display

Error Number 4

An overflow was forced (+ to -), but when tested with SNO the Skip occurred.

Error Display: No data display

Error Number 5

After executing SNO instruction, another one was executed and no Skip occurred.

Error Display: No data display

Error Number 6

An overflow was forced (- to +), but when tested with SNO the Skip occurred.

Error Display: No data display

Section 21 MULTIPLY INTEGER

Error Number 1

The simulation of the MUI instruction was accomplished by adding and shifting.

Error Display: A = simulated results, Q = simulated results (most significant)

A = instruction results, Q = instruction results (most significant)

Section 22 DIVIDE INTEGER

Error Number 1

The simulation of the DIV instruction was accomplished by subtracting and shifting.

Error Display: A = simulated quotient, Q = simulated remainder
A = instruction quotient, Q = instruction remainder

Error Number 2

An overflow occurred but the OVERFLOW indicator was not set.

Error Display: Same as Error Number 1

Error Number 3

No overflow occurred but the OVERFLOW indicator was set.

Error Display: Same as Error Number 1

Section 23 STORE PARITY TO A

Errors Number 1 and 2

The SPA instruction is simulated and the results of the simulation are compared to the instruction results.

Error Display: A = parity of simulation, Q = parity of instruction
A = contents of A when parity was determined
Q = data stored by SPA

Section 24 SET TO ONES A, Q

Error Number 1

The SET instruction is executed and then A and Q are checked for all bits being set.

Error Display: A = correct data = A and Q, Q = contents of A after SET execution
A = contents of Q after SET execution

Section 25 CLEAR TO ZERO A, Q

Error Number 1

The CLR instruction is executed and then checked for all bits being cleared in AQ.

Error Display: A = correct data = A and Q, Q = contents of A after CLR execution
A = contents of Q after CLR execution

Section 26 TRANSFER A TO A, Q

Error Number 1

The TRA instruction is executed and then the registers are checked for equality with the original contents of A.

Error Display: A = correct data, Q = contents of A after TRA execution
A = contents of Q after TRA execution

Section 27 TRANSFER Q TO A, Q

Error Number 1

The TRQ instruction is executed and then the destination register is checked for equality with the original contents of Q.

Error Display: A = correct data, Q = contents of Q after TRQ execution

Section 28 TRANSFER COMPLEMENT A TO A, Q

Error Number 1

The TCA instruction is simulated by complementing the operand with an EOR instruction. The results of the simulation are then compared to the destination registers of the TCA instruction.

Error Display: A = simulated data, Q = contents of A after TCA
Q = contents of Q after TCA

Section 29 TRANSFER COMPLEMENT Q TO A, Q

Error Number 1

The TCQ instruction is simulated by complementing the operand with an EOR instruction. The results of the simulation is then compared to the destination registers of the TCQ instruction.

Error Display: A = simulated results, Q = contents of A after TCQ
A = contents of Q after TCQ execution

Section 2A TRANSFER THE ARITHMETIC SUM A, Q TO A, Q

Error Number 1

The ADD instruction is used to simulate the AAQ instruction.

Error Display: A = simulated results, Q = contents of A after AAQ execution
A = contents of Q after AAQ execution

Section 2B TRANSFER EXCLUSIVE OR A, Q TO A, Q

Error Number 1

The EOR instruction is used to simulate the EAQ instruction.

Error Display: A = simulated results, Q = contents of A after EAQ execution
A = contents of Q after EAQ execution

Section 2C TRANSFER THE LOGICAL PRODUCT OF A, Q, TO A, Q

Error Number 1

The AND instruction is used to simulate the LAQ instruction.

Error Display: A = simulated results, Q = contents of A after LAQ execution
A = contents of Q after LAQ execution

Section 2D TRANSFER COMPLEMENT LOGICAL PRODUCT A, Q, TO A, Q

Error Number 1

The AND and FOR instructions are used to simulate a CAQ instruction.

Error Display: A = simulated results, Q = contents of A after CAQ execution
A = contents of Q after CAQ execution

Section 2E SET TO ONES - M

Error Number 1

The SET instruction is executed and then M is checked for all bits being set.

Error Display: A = simulated results, Q = contents of M after SET execution

Section 2F TRANSFER A TO M

Error Number 1

Simulation accomplished the same as Section 26.

Error Display: A = correct data, Q = contents of M after TRA

Section 30 CLEAR TO ZERO - M

Error Number 1

The CLR instruction is executed and then M is checked for all bits to be cleared.

Error Display: A = correct data, Q = contents of M after SET

Section 31 TRANSFER Q TO M

Error Number 1

Simulation accomplished the same as Section 27.

Error Display: A = correct data, Q = contents of M after TRQ

Section 32 TRANSFER COMPLEMENT A TO M

Error Number 1

Simulation accomplished the same as Section 28.

Error Display: A = correct data, Q = contents of M after TCA

Section 33 TRANSFER COMPLEMENT Q TO M

Error Number 1

Simulation accomplished the same as Section 29.

Error Display: A = correct data, Q = contents of M after TCQ

Section 34 TRANSFER ARITHMETIC SUM A, Q TO M

Error Number 1

Simulation accomplished the same as Section 2A.

Error Display: A = simulated results, Q = contents of M after AAQ

Section 35 TRANSFER EXCLUSIVE OR A, Q TO M

Error Number 1

Simulation accomplished the same as Section 2B.

Error Display: A = simulated results, Q = contents of M after EAQ

Section 36 TRANSFER THE LOGICAL PRODUCT OF A, Q TO M

Error Number 1

Simulation accomplished the same as Section 2C.

Error Display: A = simulated results, Q = contents of M after LAQ

Section 37 TRANSFER COMPLEMENT LOGICAL PRODUCT A, Q TO M

Error Number 1

Simulation accomplished the same as Section 2D.

Error Display: A = simulated results, Q = contents of M after CAQ

Section 38 TRANSFER M TO A, Q, M

Error Number 1

The TRM instruction is executed and then the registers are checked for equality with original contents of M.

Error Display: A = simulated results, Q = contents of A after TRM
A = contents of Q after TRM, Q = contents of M after TRM

Section 3A TRANSFER COMPLEMENT M TO A, Q, M

Error Number 1

The TCM instruction is simulated by complementing the operand with an EOR instruction.

Error Display: A = simulated results, Q = contents of A after TCM
A = contents of Q after TCM, Q = contents of M after TCM

Section 3B TRANSFER COMPLEMENT Q + M TO A, Q, M

Error Number 1

The TCB instruction is simulated by using AND and EOR instructions.

Error Display: A = simulated results, Q = contents of A after TCB
A = contents of Q after TCB, Q = contents of M after TCB

Section 3C TRANSFER ARITHMETIC SUM A, M TO A, Q, M

Error Number 1

The AAM instruction is simulated by using the ADD instruction.

Error Display: A = simulated results, Q = contents of A after AAM
A = contents of Q after AAM, Q = contents of M after AAM

Section 3D TRANSFER ARITHMETIC SUM A, Q + M TO A, Q, M

Error Number 1

The AAB instruction is simulated by using ADD, EOR, and AND instructions.

Error Display: A = simulated results, Q = contents of A after AAM
A = contents of A after AAB, Q = contents of M after AAB.

Section 3E TRANSFER EXCLUSIVE OR A, M TO A, Q, M

Error Number 1

The EAM instruction is simulated by using the ROR instruction.

Error Display: A = simulated results, Q = contents of A after EAM
A = contents of Q after EAM, Q = contents of M after EAM

Section 3F TRANSFER EXCLUSIVE OR A, Q + M TO A, Q, M

Error Number 1

The EAB instruction is simulated by using the AND, ADD, and EOR instruction.

Error Display: A = simulated results, Q = contents of A after EAB
A = contents of Q after EAB, Q = contents of M after EAB

Section 40 TRANSFER LOGICAL PRODUCT A, M TO A, Q, M

Error Number 1

The LAM instruction is simulated by using the AND instruction.

Error Display: A = simulated results, Q = contents of A after LAM
A = contents of Q after LAM, Q = contents of M after LAM

Section 41 TRANSFER LOGICAL PRODUCT A, Q + M TO A, Q, M

Error Number 1

The LAB instruction is simulated by using the AND, ADD, and EOR instructions.

Error Display: A = simulated results, Q = contents of M after LAB
A = contents of Q after LAB, Q = contents of M after LAB

Section 42 TRANSFER COMPLEMENT LOGICAL PRODUCT A, M TO A, Q, M

Error Number 1

The CAM instruction is simulated by using AND and EOR instructions.

Error Display: A = simulated results, Q = contents of A after CAM
A = contents of Q after CAM, Q = contents of M after CAM

Section 43 TRANSFER COMPLEMENT LOGICAL PRODUCT A, Q+M TO A, Q, M

Error Number 1

The CAB instruction is simulated by using AND, ADD, and EOR instructions.

Error Display: A = simulated results, Q = contents of A after CAB
A = contents of Q after CAB, Q = contents of A after CAB

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENT - About 2500_{10} locations.

B. TIMING - 1 min. 15 sec.

C. EQUIPMENT CONFIGURATION

1. 1704 with a minimum of 4K of memory.

2. A device for loading the program.

1700 MEMORY TEST
MEM014 Test No. 14
(CP = 23)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Bit 9 of the A register on the second parameter stop must be set to test a specific area of core by entering the test block addresses in the third parameter stop.

2. The operator should never restart SMM while the Memory Test is testing bank zero, since SMM and the Memory Test have been moved to the last memory stack.

3. This test is valid in core memory only. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. The Memory Test operates under control of the 1700 System Maintenance Monitor (SMM17).

2. The calling sequence is that specified by SMM17.

C. PARAMETERS

1. Normal operation requires no parameters. Preselected test sections to be run are Sections 02 to 40.

2. To alter or review parameters, use the procedure explained in SMM17 Description.

a. FIRST STOP

A = Stop ID words = $1441
Q = Tests Stop/Jump word

b. SECOND STOP

A =

15 8	7 0
Test Control Bits	Sections to be Run

Bit 0 Section 01 - Single Cell Inspection

Bit 1 Section 02 - All ones - All zeros

Bit 2 Section 04 - S Register Test

Bit 3 Section 08 - Worst Pattern

Bit 4 Section 10 - A-5 Pattern

Bit 5 Section 20 - Random Pattern

Bit 6 Section 40 - Special Pattern

Bit 7 Not used

Bit 8 Move the test to the lower half of each stack and test the remaining half of the stack with all sections selected.

Bit 9 Test only the fixed test block, as entered in stop 3 with all sections selected.

Bit 10 to 15 Not used

Q =

15 0
Test Pattern for Sections 01 and 40

Allows the operator to test core with any special pattern.

c. THIRD STOP

A =

15 0
First Word Address of the Fixed Test Block

Q =

15 0
Last Word Address + 1 of the Fixed Test Block

The fixed test block must be $\geq 40_{16}$ (64_{10}) words and the lower 6 bits of the first and last word addresses must be cleared. Also bit 9 in the control word must be set to inform the Memory Test to use the fixed test block.

d. FOURTH STOP

```
A = 15                                    0
┌─────────────────────────────────────────┐
│             Number of Reads             │
└─────────────────────────────────────────┘
Q = 15                                    0
┌─────────────────────────────────────────┐
│                Not Used                 │
└─────────────────────────────────────────┘
```

Enter in A the number of times each test cell in the Worst Pattern section will be read before the data is checked.

D. SELECTIVE SKIP and STOP SETTINGS

1. Stop - for information stops.

2. Skip - forces SMM Parameter stop.

3. Stop/Jump Parameter

   a. Bits 0 to 10 - refer to SMM17 for standard usage (V. B).

   b. Bit 11 - repeats all test sections selected in the same test block (fixed test block on the same memory stack).

   c. Bit 12 - output errors only at end of each section pass with Error stop CXXX and Omit Error stops 8XXX and 4XXX.

E. MESSAGES

1. Typeouts

   a. Normal Program Typeouts

      1) Memory Test identification at start of test.

```
MEMORY TEST
IA = ----, FC = --      CP23, VER. 3.1
```

      2) "CLEAR PROTECT SWITCH. DISABLE AUTOLOAD PROTECT" followed by programmed stop. Run when action complete. (If key up, monitor error stop for Protect fault will occur instead.)

      3) End of Test (see SMM17 Description V. A).

```
A        Q        A            Q

1424     S/J      PASS NO.     RETURN ADDRESS
```

   b. Error typeouts are in the standard SMM17 format (see SMM17, V. A).

2. Error Codes

a. Error 1 - A parity error was sensed after the cell under test was read three times. If the pattern read equals the pattern stored in the test cell, the error occurred in the set up of the pattern and not in cell under test.

b. Error 2 - The pattern of the third read does not equal the pattern of the second read.

c. Error 3 - The pattern of the third read does not equal the pattern of the first read.

d. Error 4 - The pattern read from the test cell does not equal the pattern stored in the cell.

e. Error 0 - The pattern stored in the test cell of Section 01, single cell inspection, does not equal the pattern read out.

F. ERROR STOPS

Three different Memory Test error formats can be identified on the first stop by the number of stops designated in the identification word (see SMM17 Description, SMM/Operator Interface). A fourth error format is identified only with the single cell inspection section.

1. First Error in Group (Five Stops)

This error stop is in the Common Compare subroutine used by sections 02-40. This format displays the first error of a group of errors that have occurred in consecutive core locations.

a. FIRST STOP

A = 1458 Identification Word.
Q = XXX8 Stop/Jump parameter.

b. SECOND STOP

A = section number/error code
Q = return address of where the stop originated.

c. THIRD STOP

A = expected pattern (pattern stored in the test cell)
Q = actual pattern read out

d. FOURTH STOP

A = address of the failing test cell
Q = number of failing cells so far detected in this section

e. FIFTH STOP

A = common address bits of all failing cells so far detected in this section that are set ("1's")

Q = common address bits of all failing cells so far detected in this section that are reset ("0's")

2. Last Error of Group (Three Stops)

This error stop is in the Common Compare subroutine used by sections 02-40. This format will occur after the five-stop format (see Error Stops preceding) has occurred and two or more errors in consecutive core locations have been detected.

a. FIRST STOP

A = 1438 Identification Word.

Q = XXX8 Stop/Jump parameter.

b. SECOND STOP

A = section number/error code

Q = return address of where the stop originated.

c. THIRD STOP

A = address of the first test cell that was read correctly after two or more errors were detected in consecutive core locations.

Q = not used.

3. Section Error Summary (Four Stops)

This error stop is in the Common Compare subroutine used by sections 02-40. This format will occur at the end of sections 02-40 whenever any errors are detected within the test section. There may be two four-stop formats in the same section since some sections used one pattern on the first pass and its complement pattern on the second pass. (Example: All "1's" the first pass, all "0's" the second pass.)

a. FIRST STOP

A = 1448 Identification Word.
Q = XXX8 Stop/Jump parameter.

b. SECOND STOP

A = section number/error code
Q = return address of where the stop originated.

c. THIRD STOP

A = common address bits of all failing cells detected within this section that were set "0's".

Q = common address bits of all failing cells detected within this section that were reset ("0's").

d. FOURTH STOP

A = number of failing cells detected in this section.

Q = 0000 this is the first pass of the section.

Q = FFFF this is the second pass of the section using the complement pattern of the first.

Example of Previous Three Error Formats

The Memory Test detects parity errors in the test cells located at addresses 1031$_{16}$, 1032, 1033, and 1034 in the test block from locations 1000$_{16}$ to 2000$_{16}$. The errors were found after the memory worst pattern in Section 08 was stored in the test block and then read back.

The Memory Test will display the following information:

1) After the first parity error at location 1031$_{16}$ is detected:

A = 1458	Q = XXX8
A = 0301	Q = XXXX
A = 0000	Q = XXXX
A = 1031	Q = 0001
A = 1031	Q = EFCE

2) After location 1035$_{16}$ is read out correctly:

A = 1438	Q = XXX8	(Three stops identifies last error of group.)
A = 0301	Q = XXXX	
A = 1035	Q = FFFF	(Address 1034 was the last failure in the group; 1035 was the first correct memory address.)

3) After this pass of the section is completed:

A = 1448	Q = XXX8	(Four stops identifies the section error summary.)
A = 0301	Q = XXXX	
A = 1030	Q = EFC8	
A = 0004	Q = 0000	

4. Single Cell Inspection Error

This error stop is only used on the Run-Alone Single Cell Inspection Test Section. If this test section has been selected, only the following error format can occur:

a. FIRST STOP

A = 1448 Identification Word.
Q = XXX8 Stop/Jump parameter.

b. SECOND STOP

A = section number 01/error code 00
Q = return address of where the stop originated.

c. THIRD STOP

A = expected pattern (pattern stored in test cell)
Q = actual pattern read out

d. FOURTH STOP

A = address of the test cell
Q = not used

II. DESCRIPTION

The test determines the memory size and checks memory in 1000_{16} word blocks (one stack). The first test block will be the last memory stack and testing will continue from higher to lower stacks until only stack zero is left to be tested. The test is moved to the last stack and stack zero is then tested. If the machine was only 4K of memory only the upper half of the stack will be tested.

The test consists of six main body sections and one optional section. The first two sections consist of a simple check of memory for the ability to hold all ones, all zeros and to hold its own cell address. The third section, worst pattern, will test memory for the ability to hold worst pattern and complement worst pattern.

Sections 10, 20, and 40 test memory to hold $AAAA_{16}$ 5555_{16} pattern, random pattern, and special pattern; the operator has the option to change the special pattern at parameter input time. In all sections except Section 4, the program protect plane is set equal to bit 15 of each word of the test block.

Section 2 - ALL "1's", ALL "0's"

A. Set the test block to all "1's".

B. Each test cell is excited by reading it three times.

C. Data is checked from read three.

D. Complement the pattern in the test block and repeat steps B and C.

Section 4 - S REGISTER TEST

A. Set each cell within the test block to its own address.

B. Each test cell is excited by reading it three times.

C. Data is checked from read three.

Section 8 - WORST PATTERN

A. Set the test block to worst pattern.

B. The pattern of the test cell is determined by the cell address and this pattern is complemented and stored in the test cell.

C. The test cell is then excited in two ways:

1. An all-zeros cell within the inhibit group of the test cell is found and then read X number of times, which may be set at parameter input time.

2. The test cell is excited by reading it three times.

D. Data is checked from read three.

E. Recomplement pattern in test cell.

F. Set the test block to the complement worst pattern and repeat steps B, C, D and E.

Section 10 - $AAAA_{16}$, 5555_{16} PATTERNS

A. Set the test block to $AAAA_{16}$, 5555_{16}.

B. Each test cell is excited by reading it three times.

C. Data is checked from read three.

D. Complement the pattern in the test block and repeat steps B and C.

Section 20 - RANDOM PATTERN

A. Generate eight random patterns.

B. Set the test block to these patterns.

C. Determine test cell pattern by the cell address.

D. Each test cell is excited by reading it three times.

E. Data is checked from read three.

F. Complement the pattern in the test block and repeat steps C, D and E.

Section 40 - SPECIAL PATTERN

A. Set the test block to the pattern entered on stop 2 in the Q Register at parameter input time.

B. Each test cell is excited by reading it three times.

C. Data is checked from the read three.

D. Complement the pattern in the test block and repeat steps B and C.

Section 1 - SINGLE CELL INSPECTION

This section is an optional section and may only be manually selected by the operator at parameter input time. At that time the address of the cell to be checked must be entered in the A register on the third stop.

A. Store pattern in test cell.

B. Read test cell.

C. Check data.

Notes: Testing specific core areas: At parameter input time the operator may select any specific core area by setting bit 9 in the A register on the second stop and entering the first and last word address + 1 in the A and Q registers on the third stop. If the block of core to be tested is greater than the core available, the block will be tested in two passes.

After the Memory Test has run all sections selected, it will determine which device SMM17 is loading from, and then move a hand bootstrap loader for that device to the end of core. The Memory Test will also clear all the memory protect bits and then protect only the hand loader. The starting address for all hand bootstrap loaders is the last core location - 1.

Worst Pattern is defined as follows:

A. Locations 00_{16} to 03_{16} are set to $AAAA_{16}$, 0000_{16}, 5555_{16}, $FFFF_{16}$ pattern which is repeated to location $3F_{16}$.

B. Locations 40_{16} to 43_{16} are set to $FFFF_{16}$, 5555_{16}, 0000_{16}, $AAAA_{16}$ pattern which is repeated to location $7F_{16}$.

C. Locations 80_{16} to BF_{16} are set equal to the complement of the first 40_{16} locations.

D. Locations $C0_{16}$ to FF_{16} are set equal to the complement of location 40_{16} to $7F_{16}$.

E. This pattern is repeated every 100_{16} locations throughout the stack. Hence, storage is set to worst pattern (see Figure 1).

III. PHYSICAL REQUIREMENTS

A. MEMORY REQUIREMENT about 3800_{10} locations and all remaining memory.

B. TIMING - 1 min. 30 sec.

C. EQUIPMENT CONFIGURATION - 1704 Computer with 4K memory and device for loading Memory Test and SMM.

	00	01	02	03	04	3C	3D	3E	3F
000	AAAA	0000	5555	FFFF	AAAA	AAAA	0000	5555	FFFF
040	FFFF	5555	0000	AAAA	FFFF		5555	0000	AAAA
080	5555	FFFF	AAAA	0000	5555			AAAA	0000
OCO	0000	AAAA	FFFF	5555	0000				5555
100	AAAA	0000	5555	FFFF					
140	FFFF	5555	0000						
180	5555	FFFF							
ICO	0000								
F00	AAAA								FFFF
F40	FFFF	5555						0000	AAAA
F80	5555	FFFF	AAAA				FFFF	AAAA	0000
FCO	0000	AAAA	FFFF	FFFF	5555	000	AAAA	FFFF	5555

Figure 1. Schematic Representation of 4096 Words
1700 Memory's Worst Pattern

1700 PROTECT TEST
(RPTOQ9 Test No. 09)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. A flashing OVERFLOW light is used as a signal to the operator.

   a. When the test is being initialized, the flashing light means the operator should clear the STOP switch and set the PROGRAM PROTECT switch. A message will be typed out at this time if a teletype is available.

   b. While the test is running, the flashing light means the operator should set the STOP switch, clear the Q register, and run the computer. An End of Section or an End of Test stop will then occur. The STOP switch should again be cleared before proceeding to the next section or before repeating the test.

   c. If the Repeat Section bit is set, an End of Section stop will occur each time the test is repeated.

2. There is no signal at the end of test if the corresponding stop bit is not set in the test's Stop/Jump parameter. Thus, system automation is an operator option; he can set the bit and wait completion to set the STOP switch or use the ensuing load, test heading typeout, or "Build Test List" typeout (a function or RPT's test list position, see B. 3) to key operator action.

B. LOADING PROCEDURE

1. The Protect Test operates under control of the 1700 System Maintenance Monitor (SMM17).

2. The calling sequence is that specified by SMM17.

3. The RPT acts as a test list load stringer (see SMM17, IV. A. 2) and awaits the completion of those loaded before it, then runs alone.

C. PARAMETERS

1. Normal operation requires no parameters.

2. To alter or review parameters, use the procedure explained in SMM17 (see SMM17 Description, VA).

a. First Stop

A = 15 0

IDENTIFICATION WORD

Q = 15 0

STOP/JUMP PARAMETER

b. Second Stop

A = 15 0

SELECTABLE SECTIONS

Bit	Section
0	0
1	1
2	2

Q = Inapplicable

D. SELECTIVE SKIP AND STOP SETTINGS

1. SKIP switch is cleared.
2. STOP switch is cleared except when needed for End of Section or End of Test stops.

E. MESSAGES

Typeouts

1. Normal Program Typeouts.

a. Test identification at start.

```
PROTECT TEST 9
IA = XXXX
```

(XXXX = Starting address of test.)

b. CLEAR STOP SWITCH, SET PROTECT SWITCH

(The test will hang up with the OVERFLOW light flashing until the PROTECT switch is set.)

c. End of Test

A	Q	A	Q
0924	ST/JP	PASS COUNT	RETURN ADDRESS

2. Error Stops

1st Stop	A	Q
	ID	ST/JP
2nd Stop	A	Q
	SECTION NUMBER/ ERROR CODE	RETURN ADDRESS
3rd Stop	A	Q
	FAILING INSTRUCTION	FAILING ADDRESS (SECTION 3)
4th Stop	A	Q
	NUMBER OR ERRORS	ZERO

F. ERROR CODES

Code	Description
1	Memory parity error.
F	Protect fault failed to occur.

II. DESCRIPTION

This test checks the protect hardware of the 1704 Computer. The test protects all locations in memory, performs the Parameter stop, and then waits, with the OVERFLOW light flashing, until the PROTECT switch is set. The STOP switch is also to be cleared at this time. If it is not cleared, a stop will occur on each protect fault.

A. INITIALIZATION (INIT1)

1. Multiplex until other tests in core are finished.
2. Type out test heading, initial address (IA), and frequency count (FC).
3. Set protect bits of all memory locations.

4. Enter parameters if selected.

5. Type out message to operator:

   CLEAR STOP SWITCH, SET PROGRAM PROTECT SWITCH

6. Flash OVERFLOW light and wait until PROTECT switch is set.

7. Begin execution.

B. SECTION ZERO (SECO)

This section checks for protect faults caused by the execution of non-protected EIN, IIN, SPB, CPB, EXI, and all mask register instructions.

1. Store section number.
2. Set instruction counter.
3. Clear Protect bit of execution address and address following.
4. Set mask for internal interrupt and enable interrupts.
5. Set counter. This is the number of times each instruction will be executed ($F000_{16}$).
6. Clear Error Counter.
7. Load instruction and store it in the execution address and in the error routine.
8. Execute instruction.
9. Check for a protect fault. If it did not occur, add one to error count.
10. Check loop counter. If not zero, repeat from item 8.
11. If loop count is zero, check error count. If not zero, make an error stop and output the failing instruction and the number of errors.
12. If no errors occurred, update instruction counter and loop to item 11 until all instructions have been set.
13. Return to control routine (CNT).

C. SECTION ONE (SEC1)

This section checks for protect faults caused by trying to execute a protected instruction following the execution of a non-protected instruction.

1. Store section number.
2. Set Protect bits in all memory locations.

3. Set instruction counter.

4. Clear Protect bit on execution address -1.

5. Set mask for internal interrupt and enable interrupts.

6. Set loop counter. The number of times each instruction will be executed ($F000_{16}$).

7. Load instruction and store it in execution address and in the error routine.

8. Execute a non-protected no-op instruction and then the instruction being checked.

9. Check for protect fault interrupt.

10. If protect fault did not occur, add one to error count.

11. Loop to item 7 until instruction has been executed $F000_{16}$ times.

12. Check for errors (protect fault did not occur); if present, make an error stop and display the failing instruction and the number of errors.

13. Loop to item 6 until all instructions have been checked.

14. Return to control routine (CNT).

D. SECTION TWO (SEC2)

This section generates protect faults by attempting to store in all available memory locations (protected) with a non-protected instruction.

1. Store section number.

2. Set protect bits in all available memory locations.

3. Set instruction counter.

4. Clear Protect bit of instruction execution address.

5. Set mask for internal interrupt and enable interrupts.

6. Pick up last word address of available memory.

7. Form the instruction to be executed and store it in the execution address and in the error routine.

8. Execute the instruction.

9. Check for protect fault. If it did not occur, make an error stop and display the failing instruction and the failing address.

10. If protect fault occurred, subtract one from the memory storage address.

11. Loop to item 8 until all available memory has been checked.

12. Update instruction counter. If not zero, loop to item 6.

13. Return to control routine (CNT).

E. CONTROL ROUTINE (CNT)

This routine checks for section selection, re-entry of parameters, End of Section and End of Test stop, and for Repeat Section or Repeat Test.

1. Check for re-entry of parameters.
2. Check for section 0 selection. If not selected, go to item 6.
3. Run section 0.
4. Check for End of Section stop.
5. Check for Repeat Section.
6. Check for section 1 selection. If not selected, go to item 10.
7. Run section 1.
8. Check for End of Section stop.
9. Check for Repeat Section.
10. Check for section 2 selection. If not selected, go to item 14.
11. Run section 2.
12. Check for End of Section stop.
13. Check for Repeat Section.
14. Check for End of Test stop.
15. Check for Repeat Test.
16. Return to SMM. If pass count is not zero, the test will be repeated.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS

Approximately 500 locations.

B. TIMING - 0 min. 45 sec.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 4K memory.
2. A device for loading the program.

1774 SYSTEM CONTROLLER COMMAND TEST
(CAR01B Test No. 1B)
(CP = 02)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. If the operator wishes to run section 5 or 6, he must call the system controller command test alone. The test overlays the normal interrupt processor with its own special processor in these sections.

2. Section 5 may be run only if the system controller has an AQ channel. A special interrupt generator, which is attached to the AQ channel, is used to generate interrupts on the lines corresponding to bits set in Q.

   a. Special Interrupt Generator

      The INT special cables consist of a male 61-pin connector and 15 female 3-pin interrupt jacks. Each Q bit (bit 1 to bit 15) is wired via 3-wire interrupt cable from the 61-pin connector to the 3-pin interrupt jacks. The special interrupt cable allows each Q bit (excluding bit 0) to generate the corresponding interrupt level.

3. If sections 5 or 6 are selected, the test must be executed to completion. This is to allow restoring of the normal interrupt processor for additional tests that are to be executed.

B. LOADING PROCEDURE

This test is called as test number 1B under SMM17. It may be run with other tests if (and only if) section 5 is not chosen and section 6 is not chosen.

C. PARAMETERS

1. Normal operation runs test sections 1 through 4 and 6 and requires no parameters.

2. To alter parameters, use the procedures outlined in SMM17. The identification word and Stop/Jump parameter are displayed on the first stop. On the second stop, set bit 0 to run section 1, etc., according to the table below:

Bit in A	Test Section
0	Section 1 -- LDA, Character mode, with character designation bit = 0.
1	Section 2 -- LDA, Character mode, with character designation bit = 1. Indexing used here.
2	Section 3 -- STA, Character mode, with character designation bit =0. Indexing used here.

Bit in A	Test Section
3	Section 4 -- STA, Character mode, with character designation bit = 1.
4	Section 5 -- Character mode enable/disable with interrupts.
5	Section 6 -- powerfail interrupt on line 0 and auto restart test.

II. MESSAGES

A. NORMAL

1. Test identification at beginning of test:

    CAR01B, SYSTEM CONTROLLER TEST

2. End of test:

A	Q	A	Q
1B04	STOP	PASS NO.	RETURN ADDRESS

3. Message to operator to hook up the interrupt generator to the AQ channel (section 5):

    INTP. GEN. ON AQ CHAN/

4. Message to operator that he may restore the AQ channel as it originally was (section 5):

    RESTORE AQ CHAN.

5. Message to operator to set AUTO RESTART switch:

    SET AUTO RESTART SWITCH

6. Message to operator to drop mainframe power, then bring it up again, checking Auto Restart operation:

    DROP POWER, THEN RESTORE

7. Message to operator that he may clear the AUTO RESTART switch:

    CLR AUTO RESTART SWITCH

B. ERROR

The following is typed out:

1. Identification word
2. Stop/Jump parameter
3. Section and error number
4. Return address
5. Actual results
6. Expected results

C. ERROR STOPS

Error Code	Description
01	LDA, character failed with character designator bit = 0. (A) = actual contents of A upon the LDA character instruction (Q) = expected contents of A upon the LDA character instruction
02	Same as 01 above, except the character designator bit here = 1.
03	STA, character failed with character designator bit = 0. (A) = actual data stored in memory by the STA character instruction (Q) = expected data stored in memory by the STA character instruction
04	Same as 03 above, except the character designator bit here = 1.
05	Level 0 - No interrupt occurred when power was dropped on the mainframe.
06	Level 0 - An interrupt occurred when power was dropped, but skip on parity error indicated a parity error caused the interrupt.
07	Level 0 - An interrupt occurred when power was dropped, but skip on protect fault indicated a protect fault caused the interrupt.

Error Code	Description
11 - 1F	Character mode was not properly re-enabled upon exit from interrupt state. Error 11 corresponds to interrupt state 1, 12 corresponds to state 2, etc.
21 - 2F	Character mode was not properly disabled when an interrupt occurred on the line indicated. Error 21 corresponds to state 1, 22 corresponds to state 2, etc.

III. DESCRIPTION

A. Section 1 - LDA, Character mode is checked with the character designator bit = 0 (upper half of memory location referenced). The EOR instruction is used in testing proper operation.

B. Section 2 - LDA, Character mode is checked with the character designator bit = 1. Indexing is used, with the index originally set = FFFF. The index is then decremented by twos, checking proper right shift and sign extension of the address in I.

C. Section 3 - STA, Character mode is checked with the character designator bit = 0. Indexing is used, with the index originally set = FFFE. The index is then decremented by twos, checking proper right shift and sign extension of the address in I.

D. Section 4 - STA, Character mode is checked with the character designator bit = 1.

E. Section 5 - The automatic disabling of Character mode is checked upon an interrupt on each interrupt line (except 0). The automatic re-enabling of Character mode is checked upon execution of exit state for each interrupt state. This requires the operator to hook up the AQ interrupt generator, and, upon completion, to remove it.

F. Section 6 - Test the auto restart feature, checking that the proper level of interrupt is generated when power is dropped on the mainframe. The operator is instructed to set the AUTO RESTART switch, turn off power and turn it back on again, and clear the AUTO RESTART switch. Unlike lines 1-F tested in section 5, this is tested only once (unless Repeat Condition is set).

NOTE

The system controller test must be called to run alone if section 5 is to be run. Section 5 can only be run:

a) If the system controller has an AQ channel.

b) If a special interrupt generator is available which is attached to the AQ channel, and which generates an interrupt on each line corresponding to a bit set in Q.

The system controller test must be called to run alone if section 6 is to be run.

IV. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS

Approximately 900_{10} locations.

B. TIMING -

C. EQUIPMENT CONFIGURATION

1. 1774 SC with 4K memory.
2. A device for loading the program.

1700 MEMORY TEST
(MY2002 Test No. 02)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. The test requires a machine with at least 8K (two stacks) of memory.
2. The operator never should restart SMM while the test is testing stack zero, since SMM and the test have been moved to the stack specified by the test parameters (stack one is standard).
3. To test 65K, SMM must have been loaded in 65K mode.
4. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

The test is loaded under SMM. When the test is given control to utilize, it does not return control to SMM until the test is completed. When control is returned to SMM, the next test is loaded and this test is destroyed.

C. PARAMETERS

1. Normal operation requires no parameters. Preselected test sections to be run are Sections 00 to 05.
2. To alter or review parameters, use the procedure explained in SMM17.

First Stop

A1 = ID Word ($0231)
Q1 = Stop/Jump Parameter

Second Stop

A2 = Sections to be Run (Prestored as $003F)

Bit 0 - Section 00 - Zeros
Bit 1 - Section 01 - Ones
Bit 2 - Section 02 - Address Test
Bit 3 - Section 03 - Parity Plane Test
Bit 4 - Section 04 - Worst Pattern
Bit 5 - Section 05 - First Pass Test
Bits 6 to 15 - Not used

Q2 = Stacks to be Tested (Bits 0-7)

Bit 0 - Stack 0
Bit 1 - Stack 1
Bit 2 - Stack 2

Bit 3 - Stack 3
Bit 4 - Stack 4
Bit 5 - Stack 5
Bit 6 - Stack 6
Bit 7 - Stack 7
Bit 11 - Stack 11
Bit 12 - Stack 12
Bit 13 - Stack 13
Bit 14 - Stack 14
Bit 15 - Stack 15

(Q2 is prestored as $0002)

Third Stop

A3 = time between successive Memory References in Section 05 of test (prestored as $0101) only one bit 0-3 should be set at a time (not used in 65K mode).

Bit 0 - 2.2 μsec
Bit 1 - 3.3 μsec
Bit 2 - 4.4 μsec
Bit 3 - 5.5 μsec
Bit 4 to 7 - Not used

A3 also designates the stack to which the test and SMM are moved when testing stack zero. Only one bit in bits 8-15 should be set at a time.

Bit 8 - Stack 1
Bit 9 - Stack 2
Bit 10 - Stack 3
Bit 11 - Stack 4
Bit 12 - Stack 5
Bit 13 - Stack 6
Bit 14 - Stack 7
Bit 15 - Not used

Q3 = X X X X
XXXX = Number of passes to be made with each pattern in Section 05 (prestored as $0001).

D. SELECTIVE SKIP AND STOP SETTINGS

1. STOP switch - Normal SMM usage
2. SKIP switch - Brings up test Stop/Jump parameter stop
3. Stop/Jump Parameter
   a. Bits 0 to 10 - Standard SMM17 usage
   b. Bit 11 - Isolation Bit. Loops on test cell, storing expected value and making compare check.

c. Bit 12 - Exit Section Bit. Aborts the present test section, begins operation of the next selected section. The bit is cleared by the program.

E. PROGRAM PROTECT SWITCH

The PROGRAM PROTECT switch should remain in the OFF position.

F. MESSAGES

1. Typeouts

a. Normal Program Typeouts

1) Identification at start of test.

MY2002, MEMORY TEST
IA = XXXX, FC=XX   CP2F, VER. 3.1

2) End of Section

First Stop

A = 0232	Ident Word
Q = XXXX	Stop/Jump Word

Second Stop

A = XXXX	Section
Q = XXXX	Stack Just Tested

Third Stop

A = XXXX	Number of Errors in this Section
Q = XXXX	Return Address

3) End of Test

First Stop

A = 0224	Ident Word
Q = XXXX	Stop/Jump Word

Second Stop

A = XXXX	Pass Count
Q = XXXX	Number of Errors in Test

b. Error Typeouts

Error typeouts are in the standard SMM17 format.
NOTE: All error stops consist of four stops.

First Stop

A = 0248	Identification Word
Q = XXXX	Stop/Jump Parameter

Second Stop

A = XXYY	Section/Error Code
Q = XXXX	Return address for errors 01 through 08. Location of error cell for errors 09 through 0B.

2. Error Codes

Error 1 - Compare error, no parity error was found while making the compare check.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 2 - Parity error occurred on a load instruction, but no compare error was found.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 3 - A parity error occurs while performing a load instruction. There is also a compare error.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 4 - A parity error occurs while doing a store instruction whose effective address is in the test stack. This indicates that bits were dropped in this location before the store instruction may have been properly executed. If not, a compare error will occur later.

Third Stop

A = XXXX	Value being stored
Q = XXXX	Location of error cell

Fourth Stop

A = 0000	Not used
Q = 0000	Not used

Error 5 - A parity error occurs on a load instruction which references the stack being tested. The exact error cell could not be found.

Third Stop

A = XXXX	Address of instruction being executed when error occurred or address of the check that revealed the error.
Q = 0000	Not used.

Fourth Stop

A = 0000	Not used
Q = 0000	Not used

Error 6 - A parity occurs in the Memory Stack which contains SMM and the test. This indicates that bits have been dropped within the test itself. If the operator attempts to continue the running of the test, the section in which the error occurred is aborted and the test attempts to run the next section.

Third Stop

A = XXXX	Normally gives the address stored in trapped location $100. But when stack zero is being tested, the level zero interrupt is not enabled. This stop gives the address of the parity check which revealed that there had been a parity error in the program sometime since the last parity error check.
Q = 0000	Not used

Fourth Stop

A = 0000	Not used
Q = 0000	Not used

Error 7 - Unexpected program protect fault.

Third Stop

A = XXXX Address of the location in the test stack which was being tested at the time of the error.

Q = 0000 Not used

Fourth Stop

A = 0000 Not used

Q = 0000 Not used

Error 9 - A parity error occurred while making a first pass compare check (Section 05), but no compare error is found.

Third Stop

A = XXXX Actual value of the previous pattern.

Q = XXXX Expected value of the previous pattern.

Fourth Stop

A = XXXX Actual value of the new pattern.

Q = XXXX Expected value of the new pattern

Error A - A compare error occurred while making a first pass compare check (Section 05), but no parity occurred.

Third Stop

A = XXXX Actual value of the previous pattern.

Q = XXXX Expected value of the previous pattern.

Fourth Stop

A = XXXX Actual value of the new pattern.

Q = XXXX Expected value of the new pattern.

II. DESCRIPTION

The test determines the size of memory and checks the memory in 1000_{16} word blocks (one stack). Stack one is the first to be tested. After the last stack has been tested the test and SMM are moved to the stack specified by the test parameters (normally stack one), and stack zero is then tested. The test consists of six standard sections 00 to 05. Testing is done by testing one stack with all sections before going on to the next stack.

SECTION 00 - ZEROS TEST

Tests the ability of the test stack to hold zero in every location.

A. Fill the test stack with all zeros (0000).

B. Read each location and check its contents.

SECTION 01 - ONES TEST

Tests the ability of the test stack to hold all ones in every location.

A. Fill the test stack with all ones (FFFF).

B. Read each location and check its contents.

SECTION 02 - ADDRESS TEST

Tests the S register and the ability of each cell, in the test stack, to hold its own address.

A. Starting with the first location of the test stack and continuing through the last, fill each location of the test stack with its own address.

B. Read each location and check its contents.

C. Starting with the last location of the test stack, fill each location with its own address.

D. Read each location and check its contents.

SECTION 03 - PARITY PLANE TEST

Tests the ability of each core, in the parity plane of the test stack, to hold zero and one while the rest of the plane holds worst pattern and while the rest of the plane holds complement worst pattern.

A. Fill the test stack with complement worst pattern except for plane zero which is masked to zeros. This causes worst pattern to be generated in the parity plane.

B. Disturb each location (X) and check its contents.

C. Set bit zero of location (X), disturb the location, check its contents, then replace its original contents.

D. Fill the test stack with worst pattern except for plane zero which is masked to zeros. This causes complement worst pattern to be stored in the parity plane.

E. Repeat steps B and C.

NOTE: Disturbs are done by reading a combination of locations (Y1) and (Y2), where (Y1) is a location in the pattern which contains all zeros and is in the same inhibit group as (X), but is not on a common drive line with (X), and (Y2) has the same specifications as (Y1) except that it is on a common drive line with (X). When (Y1) is read, (X) are disturbed in the Write cycle. When (Y2) is read, (X) are disturbed in the Read cycle.

SECTION 04 - WORST PATTERN TEST

Tests the ability of each location in the test stack to hold worst pattern and complement worst pattern while the remainder of the test stack holds worst pattern. Then tests the ability of each location in the test stack to hold complement worst pattern and worst pattern while the remainder of the test stack holds complement worst pattern.

A. Fill the test stack with worst pattern.

B. Disturb each location (X) and check its contents.

C. Replace each location (X) with the complement of its original contents, disturb the location, check its contents, then replace its original contents.

D. Fill the test stack with complement worst pattern and repeat steps A, B, and C.

NOTE: Disturbs are accomplished in the same manner as those in Section 03.

SECTION 05 - FIRST PASS TEST

This test burns a pattern into the test stack, and then writes a new pattern into the stack only once, and makes a compare check.

A. Make a number of passes in the test stack, storing the first pattern. On each pass the pattern is stored in each location 32 times. The number of passes is specified by the input parameters.

B. On the last pass the 32nd reference of each location, stores the second pattern in that location.

C. After this last pass, which stores a new pattern in the test stack, is completed, a compare check of each location is made.

D. Now the second pattern is burned into the test stack, the last store of the last pass stores the third pattern and another compare check is made.

E. This procedure continues until all of the patterns listed below have been used.

PATTERN 1 - All Zeros
PATTERN 2 - Worst Pattern
PATTERN 3 - All Ones
PATTERN 4 - Complement Worst Pattern
PATTERN 5 - All Zeros
PATTERN 6 - Complement Worst Pattern
PATTERN 7 - All Ones
PATTERN 8 - Worst Pattern
PATTERN 9 - All Zeros

NOTE: The time between successive references of the same location, may be varied from 2.2 microseconds to 5.5 microseconds by varying the input parameters.

III. PHYSICAL REQUIREMENTS

A. Storage Requirements

Approximately 2500_{10} locations plus SMM monitor

B. Timing

Section 0 = 1 second per stack
Section 1 = 1 second per stack
Section 2 = 2 seconds per stack
Section 3 = 3 seconds per stack
Section 4 = 3 seconds per stack
Section 5 = 20 seconds per stack
Total = 30 seconds per stack

C. Equipment Configuration

1700 Series Computer with 8K memory

1700 MEMORY TEST
(MY1012 Test No. 12)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. The test requires a machine with at least 8K (two stacks) of memory.
2. The operator never should restart SMM while the test is testing stack zero, since SMM and the test have been moved to the stack specified by the test parameters (stack one is standard).
3. To test 65K SMM must have been loaded in 65K mode.
4. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. The test is loaded under SMM. When the test is given control to utilize, it does not return control to SMM until the test is completed. When control is returned to SMM, the next test is loaded and this test is destroyed.

C. PARAMETERS

1. Normal operation requires no parameters. Preselected test sections to be run are Sections 00 and 01.
2. To alter or review parameters, use the procedure explained in SMM17.

First Stop

A1 = ID Word ($1231).
Q1 = Stop/Jump Parameter

Second Stop

A2 = Sections to be Run (Prestored as $0003)

Bit 0 - Section 00 - Worst Pattern with Multiple Indirect Referencing Disturb
Bit 1 - Section 01 - Optimal Worst Pattern Test
Bit 2 - Section 2 - Program Protect Test

Q2 = Stacks to be Tested (Bits 0-15)

Bit 0 - Stack 0
Bit 1 - Stack 1
Bit 2 - Stack 2
Bit 3 - Stack 3
Bit 4 - Stack 4
Bit 5 - Stack 5

Bit 8 - Stack 8
Bit 9 - Stack 9
Bit 10 - Stack 10
Bit 11 - Stack 11
Bit 12 - Stack 12
Bit 13 - Stack 13

Bit 6 - Stack 6
Bit 7 - Stack 7
Bit 14 - Stack 14
Bit 15 - Stack 15

(Q2 is prestored as $0102)

Third Stop

A3 designates the stack to which the test and SMM are moved when testing stack 0. Only one bit in bits 8-15 should be set at a time.

Bit 8 - Stack 1
Bit 9 - Stack 2
Bit 10 - Stack 3
Bit 11 - Stack 4
Bit 12 - Stack 5
Bit 13 - Stack 6
Bit 14 - Stack 7
Bit 15 - Not used

(A3 is prestored as $0100) (Q3 is not used)

D. SELECTIVE SKIP AND STOP SETTINGS

1. STOP switch - Normal SMM usage

2. SKIP switch - Brings up test Stop/Jump parameter stop.

3. Stop/Jump Parameter

   a. Bits 0 to 10 - Standard SMM17 usage

   b. Bit 11 - Isolation Bit. Loops on test cell, storing expected value and making compare check.

   c. Bit 12 - Exit Section Bit. Aborts the present test section, begins operation of the next selected section. The bit is cleared by the program.

E. PROGRAM PROTECT SWITCH

Normally the PROGRAM PROTECT switch should remain in the OFF position. If Section 02 is being run, the operator must set and clear the PROGRAM PROTECT switch according to the description Section 02.

F. MESSAGES

1. Typeouts

   a. Normal Program Typeouts

      1) Identification at start of test.
      MY1012, MEMORY TEST
      IA = XXXX, FC=XX, CP2F, VER. 3.1

      2) End of Section

      First Stop

A = 1232	Ident Word
Q = XXXX	Stop/Jump Word

Second Stop

A = XXXX	Section
Q = XXXX	Stack Just Tested

Third Stop

A = XXXX	Number of Errors in this Section
Q = XXXX	Return Address

3) End of Test

First Stop

A = 1224	Ident Word
Q = XXXX	Stop/Jump Word

Second Stop

A = XXXX	Pass Count
Q = XXXX	Number of Errors in Test

b. Error Typeouts

Error typeouts are in the standard SMM17 format.

NOTE: All error stops consist of four stops.

First Stop

A = 1248	Identification Word
Q = XXXX	Stop/Jump Parameter

Second Stop

A = XXYY	Section/Error Code
Q = XXXX	Return address for errors 01 through 08. Location of error cell for errors 09 through 0B.

2. Error Codes

Error 1 - Compare error, no parity error was found while making the compare check.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 2 - Parity error occurred on a load instruction, but no compare error was found.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 3 - A parity error occurs while performing a load instruction. There is also a compare error.

Third Stop

A = XXXX	Actual value
Q = XXXX	Expected value

Fourth Stop

A = XXXX	Location of error cell
Q = 0000	Not used

Error 4 - A parity error occurs while doing a store instruction whose effective address is in the test stack. This indicates that bits were dropped in this location before the store instruction may have been properly executed, if not a compare error will occur later.

Third Stop

A = XXXX	Value being stored
Q = XXXX	Location of error cell

Fourth Stop

A = 0000	Not used
Q = 0000	Not used

Error 5 - A parity error occurs on a load instruction which references the stack being tested. The exact error cell could not be found.

Third Stop

A = XXXX	Address of instruction being executed when error occurred or address of the check that revealed the error.
Q = 0000	Not used

Fourth Stop

A = 0000 Not used

Q = 0000 Not used

Error 6 - A parity occurs in the Memory Stack which contains SMM and the test. This indicates that bits have been dropped within the test itself. If the operator attempts to continue the running of the test, the section in which the error occurred is aborted and the test attempts to run the next section.

Third Stop

A = XXXX Normally gives the address stored in trapped location $100. But when stack zero is being tested, the level zero interrupt is not enabled, and this stop gives the address of the parity check which revealed that there had been a parity error in the program sometime since the last parity error check.

Q = 0000 Not used

Fourth Stop

A = 0000 Not used

Q = 0000 Not used

Error 7 - Unexpected program protect fault. This error should only occur during Section 02, since this is the only time that the PROGRAM PROTECT switch should be set.

Third Stop

A = XXXX Address of the location in the test stack which was being tested at the time of the error.

Q = 0000 Not used

Fourth Stop

A = 0000 Not used

Q = 0000 Not used

Error 8 - A program protect fault did not occur when expected.

Third Stop

A = XXXX Location being tested

Q = 0000 Not used

Fourth Stop

A = 0000	Not used
Q = 0000	Not used

Error B - A parity error occurred while making a first pass compare check (Section 00), a compare error was also found.

Third Stop

A = XXXX	Actual value of the previous pattern.
Q = XXXX	Expected value of the previous pattern.

Fourth Stop

A = XXXX	Actual value of the new pattern.
Q = XXXX	Expected value of the new pattern.

II. DESCRIPTION

The test determines the size of memory and checks the memory in 1000_{16} word blocks (one stack). Stack one is the first to be tested. After the last stack has been tested the test and SMM are moved to the stack specified by the test parameters (normally stack one), and stack zero is then tested. The test consists of two standard sections (00 and 01) and one optional section (02).

SECTION 00 - WORST PATTERN TEST WITH MULTIPLE INDIRECT REFERENCING DISTURB

Tests the ability of each core, in the parity plane of the test stack, to hold zero and one while the rest of the plane holds worst pattern and while the rest of the plane holds complement worst pattern.

A. Fill the test stack with complement worst pattern except for plane zero which is masked to zeros. This causes worst pattern to be generated in the parity plane. Skip to end of section if 65K mode.

B. Disturb each location (X) and check its contents.

C. Set bit zero of location (X), disturb the location, check its contents, then replace its original contents.

D. Fill the test stack with worst pattern except for plane zero which is masked to zeros. This causes complement worst pattern to be stored in the parity plane.

E. Repeat steps B and C.

NOTE: The disturbing of each location (X) is accomplished by setting up and initiating a multiple indirect referencing chain within the inhibit group which contains the location (X).

SECTION 01 - OPTIMAL WORST PATTERN TEST

Tests the ability of each core, in the parity plane of the test stack, to hold zero and one while the rest of the plane holds worst pattern and while the rest of the plane holds complement worst pattern.

A. Fill the test stack with complement worst pattern except for plane zero which is masked to zeros. This causes worst pattern to be generated in the parity plane.

B. Disturb each location (X) and check its contents.

C. Set bit zero of location (X), disturb the location, check its contents, then replace its original contents.

D. Fill the test stack with worst pattern except for plane zero which is masked to zeros. This causes complement worst pattern to be stored in the parity plane.

E. Repeat steps B and C.

NOTE: The disturbing of each location (X) is accomplished by reading a word containing all zeros in each inhibit group of the test stack and then by reading one of four noise producing diagonals of ones. Each location (X) in the test stack is checked with the first diagonal before the next diagonal is used. When all four diagonals have been used, complement worst pattern is stored in the test stack and the procedure is repeated.

SECTION 02 - PROGRAM PROTECT PLANE TEST

This section requires the operator to set and clear the PROGRAM PROTECT switch at specified times in the section.

A. Set the program protect bit in each location of the test stack.

B. Stop. Operator must set the PROGRAM PROTECT switch and run.

C. An attempt is made to write into each location of the test stack.

D. A check is made for expected and unexpected protect fault.

E. Stop. Operator must clear the PROGRAM PROTECT switch and run.

F. Store worst pattern in the program protect plane.

G. Repeat steps B, C, D, and E.

H. Store complement worst pattern in the program protect plane.

I. Repeat steps B, C, D, and E.

J. Clear the program protect bit in each location of the test stack.

K. Repeat steps B, C, D, and E.

III. PHYSICAL REQUIREMENTS

A. Storage Requirements
Approximately 2500_{10} locations plus SMM monitor

B. Timing

Section 0 = 9 minutes per stack in 32K mode
Section 1 = 3 minutes per stack
Section 2 = Manual operation required
Total = 12 minutes per stack

C. Equipment Configuration
1700 Series Computer with 8K memory

1723 PAPER TAPE PUNCH TEST
(PTP003 Test No. 3)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

Any time the punch Ready drops, the test will display an error diagnostic and then wait until the punch is made Ready.

Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. Call as external test under SMM17.
2. Restart test after loading by Master Clear, set P = IA and RUN.

C. PARAMETERS (in order in which they are entered)

Selected parameters are typed out.

1. CNTWRD - Control word

Bits	
14	Run test with 8-level patterns.
13	Run test with 7-level patterns.
12	Run test with 5-level patterns. (Only one bit may be set in this group.)
11=1	Run test completely in Interrupt mode.
=0	Run test in both Interrupt and Character mode.
10=1	Omit delay in Character mode.

Sections to be run if bit is set.	
7	No section.
6	No section
5	Section 20 - C9, 36 pattern
4	Section 10 - all ones, all zeros pattern
3	Section 8 - complement pyramid pattern
2	Section 4 - pyramid pattern
1	Section 2 - complement zigzag pattern
0	Section 1 - zigzag pattern

2. REPEAT - Number of times each pattern is to be repeated in each section (prestored as $1E_{16}$)

3. DELAYA - Delay constant. Increasing this number increases the switching time between Interrupt and Character mode and decreases the delay between punches. (Prestored as 0C00$_{16}$)

4. DELAYB - Delay constant. Increasing this number increases the total delay before the delay is reset. (Prestored as 0E00$_{16}$)

5. DELAYC - Delay constant. Increasing this number decreases the switching time between Interrupt and Character mode. (Prestored as 0400$_{16}$)

6. INT11 - Interrupt line mask.

D. MESSAGES

1. Test Title

   PTP003, 1723 PUNCH TEST.
   CP2F, VER. 3.1

2. IA = XXXX, FC = XX

   (XXXX = starting address of test, XX = Frequency Count)

3. End of Test Message

A	Q	A	Q
0324	S/J	PASS NO.	RETURN ADDRESS

4. Other Messages

   Comments to operator

   a. PUNCH PROTECT OFF

   b. PUNCH PROTECTED

   c. PUNCH TAPE LOW

5. Error Messages

   a. All error messages are in the format specified by SMM17.

   b. Description of Error Codes

Error Code	Description
0	(Not used)
1	External reject on a Clear Equipment function. Check equipment address for correct code.
2	Internal reject on Equipment Clear function. See error 1.
3	External reject on status. See error 1.
4	Internal reject on status. See error 1.
5	After an Equipment Clear, only Power On and Ready status bits should be set.
6	External reject on a Data Interrupt Request function.

Error Code	Description
7	Internal reject on a Data Interrupt Request function.
8	External reject on a Start Motion function.
9	Internal reject on a Start Motion function.
A	External reject on status after outputting a Start Motion function.
B	Internal reject on status after outputting a Start Motion function.
C	After outputting an Equipment Clear and Start Motion function, only Busy, Power on, Ready, and data status bits should have been set. Check status

ON ERRORS 1 TO C

The test will repeat in the same loop until the error condition is corrected On errors 5 and C, Tape Low, Alarm, and Protect status bits are not checked. The checking of these bits is done in the Punch Frame Driver (see listing tag, PPT).

D	External reject on a Clear or Request Interrupt function.
E	Internal reject on a Clear or Request Interrupt function. This error will repeat on the same loop until error condition is corrected.
F	External reject on a Stop Motion and Clear Interrupt function.
10	Internal reject on a Stop Motion and Clear Interrupt function.
11, 12	Not used.
13	External reject on status after outputting a Clear Interrupt and Stop Motion function
14	Internal reject on status after a Clear Interrupt and Stop Motion function.
15	After outputting a stop motion and Clear Interrupt function, only Power On and Ready status should have been set.

Before errors F to 15, the test is delayed 10 ms to allow the last frame to be punched and Busy status to drop. The test will repeat in the same loop until the error condition is corrected. Error 15 does not check Tape Low, Alarm, and Protect status bits.

16	External reject on a Clear Interrupt function.
17	Internal reject on a Clear Interrupt function.
18	External reject on a Data Interrupt Request function.
19	Internal reject on a Data Interrupt Request function.
1A to 1F	Not used.
20	External reject on a Clear Interrupt or Data Interrupt Request function.
21	Internal reject on a Clear Interrupt or Data Interrupt Request function.
22 and 23	Not used.
24	External reject on status just before outputting the next frame of data.
25	Internal reject on status just before outputting the next frame of data.
26	External reject on output of frame of data.
27	Interrupt status (bit 2) is set; only Data Interrupt requested, but Data Ready status is not set.
28	Interrupt bit is set. Data interrupt and alarm interrupt bits are clear.
29	Interrupt and Data Ready bits set, but no interrupts selected by program.
2A	Alarm status and Tape Break status set. The test will hang up here until the break is repaired and the punch made ready.
2B	Alarm status is set, but no alarm conditions can be found. Power on is set. Tape break and tape low are reset.

2C	Start Motion has been outputted and Ready status set, but Busy status not set.
2D	Busy bit not set.
2E	The punch has dropped Ready and will hang up here until it is made Ready again by the operator.
2F	Unidentified interrupt.
30	Alarm status is set but Power On reset. Check status.
31	External reject on status.
32	Internal reject on status.
33	External reject on Start Motion function.
34	Internal reject on Start Motion function.
35	External reject on status when entering interrupt processor, or after making punch ready and starting motion following a tape break.
36	Internal reject on status when entering interrupt processor, or after making punch ready and starting motion following a tape break.

E. ERROR STOPS

1. FIRST STOP

   (A) = ID word, (Q) = Stop/Jump parameter

2. SECOND STOP

   (A) = section number/error code (Q) = return address

3. THIRD STOP

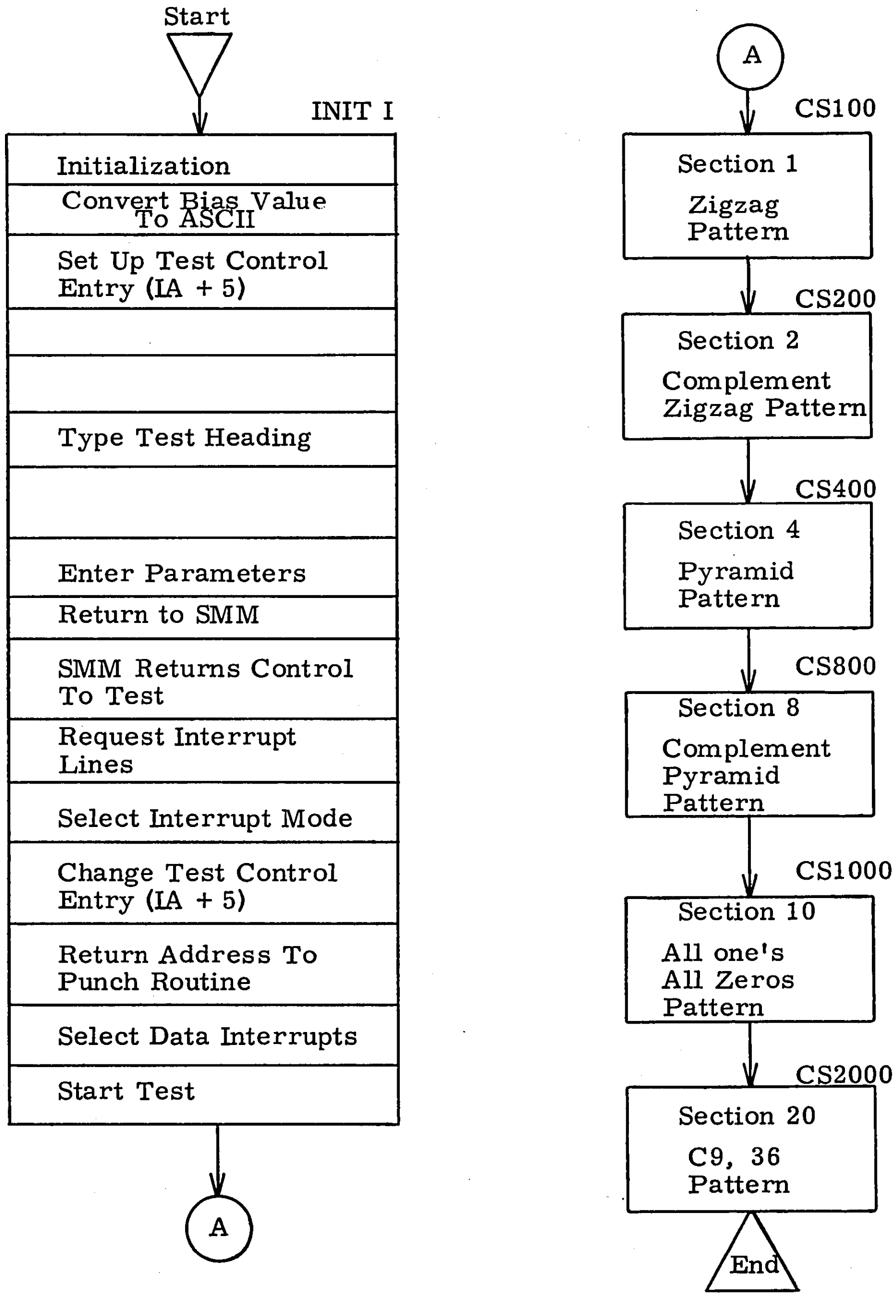
   (A) = punch status at time of error (Q) = punch equipment address

4. FOURTH STOP

   (A) = punch character (Q) = test mode at time of error
   FFFF = Interrupt mode
   0000 = Character mode

II. DESCRIPTION

A. BLOCK DIAGRAM



II. B. SECTION DESCRIPTION
(Cont'd)

1. Initialization (INIT1)
   a. Convert bias value and frequency count to ASCII and store in typeout message.
   b. Set up test return control entry (RETURN).
   c. Type test heading (INIT1B).
   d. Return to SMM.
   e. Enter parameters if selected.
   f. Return to SMM.
   g. SMM returns control to test (CNTRL).
   h. Request interrupt lines.
   i. Start punch in Interrupt mode.
   j. Change test control entry (RETURN = CHECK).
   k. Store return address (CS00A) in paper tape punch routine (PPT).
   l. Select data interrupt and exit to SMM.
   m. SMM returns control to (INTENTRY) when interrupt occurs, test ignores first interrupt and returns to (CS00A).
   n. If frequency number is greater than one, punch 10 blank frames.
   o. If not, punch one frame of all ones.
   p. Check if Section 1 is to be executed. If so, go to (CS100); if not, go to Section 2 (CS200).
2. Section 1 (CS100) - Zigzag. Pattern
   a. Set up section exit address.
   b. Check for re-entry of parameters.
   c. Determine punch level to be used (8, 7, or 5).
   d. Store words used to generate pattern.
   e. Punch section number (1 or 2) 10 times.
   f. Set up cycle counter.

g. Generate pattern.

h. Punch generated pattern.

j. Delay (in Character mode only).

k. Repeat from g until cycle is complete.

l. Repeat from g until required number of cycles has been punched.

m. Check for End of Section stop.

n. Return to Interrupt mode if test was in this mode before stop.

p. Check for repeat section.

q. Go to next section.

3. Section 2 (CS200) - Complement Zigzag Pattern

   a. If Section 2 is not selected, go to Section 4 (CS400).

   b. If selected, set up section exit address.

   c. Go to (CS100). Section 2 is a repeat of Section 1 except the complement pattern is used.

4. Section 4 (CS400) - Pyramid Pattern

   a. If Section 4 is not selected, go to Section 8 (CS800).

   b. If selected, set up section exit address.

   c. Check for re-entry of parameters.

   d. Determine punch level to be used (8, 7, or 5).

   e. Store words used to generate pattern.

   f. Punch Section 4 or 8 ten times.

   g. Set up cycle counter.

   h. Generate pattern.

   j. Punch pattern.

   k. Delay (Character mode only).

   l. Repeat from h until cycle is complete.

   m. Repeat cycle g the number of times specified by parameters.

   n. Check for End of Section stop.

   p. Return to Interrupt mode if test was in this mode before stop.

   q. Check for Repeat Section.

   r. Go to next section.

5. Section 8 (CS800) - Complement Pyramid Pattern

   a. If Section 8 is not selected, go to Section 10 (CS1000).

   b. If selected, set up section exit address.

   c. Go to (CS401). Section 8 is a repeat of Section 4 except the complement pattern is used.

6. Section 10 (CS1000) - 10 all ones and 3 all zeros

   a. If Section 10 is not selected, go to Section 20 (SC2000).

   b. If selected, omit delay between punches.

   c. Punch Section 10 ten times.

   d. Set counter for number of cycles.

   e. Generate pattern.

   f. Punch pattern.

   g. Repeat from e until cycle is complete.

   h. Update cycle count.

   j. Repeat from e until requested number of cycles has been punched.

   k. Check for End of Section stop.

   l. Return to Interrupt mode if test was in this mode before stop.

   m. Check for Repeat Section.

   n. Go to Section 20 (SC2000).

7. Section 20 (SC2000) - C9, 36 Pattern

   a. If Section 20 is not selected, go to End of Test routine (CS4000).

   b. If selected, omit delay between punches.

   c. Store word used to generate pattern.

   d. Punch Section 20 ten times.

   e. Set cycle counter.

   f. Generate pattern.

   g. Punch pattern.

   h. Repeat from f until cycle is complete.

   j. Update cycle count.

k. Repeat from f for required number of cycles.

l. Check for End of Section stop.

m. Return to Interrupt mode if test was in this mode before stop.

n. Check for Repeat Section.

p. Go to End of Test routine (CS4000).

8. Punch Routine (PPT).

a. Store (A), character to be punched.

b. Check if test is in Interrupt mode; if so, return to SMM and wait for interrupt.

c. If test is in Character mode, return to monitor.

d. When an interrupt occurs, control goes to location (INTENTR).

e. If this is the first interrupt after start motion, do not punch the character. Exit punch routine.

f. If not, check status for Ready, Data, Busy, and Power On.

g. If other than these bits are set, go to 1.

h. If status is correct, punch the character.

j. Check for repeat conditions.

k. Exit punch routine.

l. If data bit is missing, try again to punch if in Character mode (error if in Interrupt mode).

m. If data bit is present, go to CH BT and determine incorrect status.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - approximately 1400_{10} locations.

B. TIMING - 0 min. 45 sec.

C. EQUIPMENT CONFIGURATION

1. 17X4 with 4K of memory.
2. 1723/24/77 Paper Tape Punch.
3. A device for loading the program.

1721 PAPER TAPE READER TEST
(PTR004 Test No. 4)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

The frequency number of the reader test must be set to the same value used when the punch test was run, or the Repeat Test option must be used.

Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. The test operates under control of the 1700 System Maintenance Monitor (SMM17).
2. The calling sequence is that specified by SMM17.
3. Restart test after loading by Master Clear, set P = IA and RUN.

C. PARAMETERS (The selected parameters are typed out)

1. Normal operation requires no parameters.
2. To alter parameters, enter desired parameters in A and Q registers at Parameter Stops.
3. The Identification Word (A) and Stop/Jump parameter (Q) are displayed first. Next parameter displayed is the test control information (A) and Interrupt Line Mask bit (Q). This is the line that the paper tape reader interrupt is cabled to.

A = 15 10 0

TEST CONTROL BITS	NOT USED

Bits 0-9 The 1721 Paper Tape Reader test cannot select the sections to be read. These sections are selected in the 1723 Punch test, and this information is relayed to the Reader test via the punched paper tape output. (See 1723 Paper Tape Punch test).

Bit 10 = 1	Delay between frames (parameter DELRD) - character mode only - Bit 11 of control word must be zero.
Bit 10 = 0	No delay between reading frames.
Bit 11 = 1	Run test in Interrupt mode.
Bit 11 = 0	Run test in Character mode.

Q = 15

INTERRUPT LINE MASK BIT

The interrupt cable may only be changed at the start of the test or at parameter input time.

The last parameter displayed (DELRD) causes the test to delay the specified number of milliseconds between reading each frame (A) - Bit 10 of Control Word = 1, Bit 11 = 0 - preset to $000F_{16}$ (Q) inapplicable.

D. MESSAGES

1. Test title

   PTR004, 1721 READER TEST
   CP2F, VER. 3.1

2. IA = XXXX, FC = XX

   (XXXX = Starting address of test XX = Frequency Count).

3. End of Test Message

A	Q	A	Q
0424	S/J	PASS NO.	RETURN ADDRESS

4. Error Messages

   a. All error messages are in the format specified by SMM17.

   b. Description of error codes:

Error Code	Description
1	External reject on status request.
2	Internal reject on status request.
3	External reject on data input.
4	Internal reject on data input.
5	(Not used)
6	More than one bit punched in control frame.
7	All 10-level control punches not equal.

Error Code	Description
8	Control punch did not have even parity.
9	No pattern could be found for this control punch.
A	Interrupt occurred but Interrupt Status bit not set.
B	Improper status should be Ready, Busy, Power On, and Existence.
C	Alarm and/or Lost Data.
D	Unidentified interrupt.
E	Data read was not expected data.
F	First frame was not all ones.
10	Internal reject on Start Motion function.
11	External reject on Start Motion function.

E. ERROR STOPS

1. First Stop

   (A) = ID word (Q) = Stop/Jump parameter

2. Second Stop

   (A) = Section number/(Q) = Return address
   Error code

3. Third Stop (where applicable)

   (A) = Expected data (Q) = Actual data

II. DESCRIPTION

A. BLOCK DIAGRAM

See diagram for Punch test.

B. SECTION DESCRIPTION

1. Initialization (INIT)

   a. Convert bias value and frequency count to ASCII and store in typeout message.

b. Set up return address (INITB).

c. Type out test heading.

d. Return to SMM.

e. Enter parameters if selected in Stop/Jump word.

f. Set up for manual start if selected or if test was loaded from 1721 Tape Reader.

g. Set up Return Address (IA+5).

h. Return to SMM.

2. Control (CNTRL)

SMM returns control to test through Return (IA+5).

a. Start paper tape motion.

b. Read until first non-zero frame.

c. First non-zero frame must be all ones - error stop if it isn't.

d. Go to first level of test (LC010).

3. Level 1 (LC010) - Zigzag pattern

a. Check if level 1 is punched; if not, go to level 2 (LC020).

b. If punched, store level number.

c. Determine tape level to be read (8, 7, or 5).

d. Space over remaining level control punches.

e. Generate zigzag pattern.

f. Read and check one frame.

g. Repeat from (e) until level 1 is read and checked; then exit to next level.

4. Level 2 (LC020) - Complement Zigzag Pattern

a. Check if level 2 is punched; if not, go to level 3 (LC040).

b. If punched, use the level 1 routine, but read the complement zigzag pattern.

5. Level 3 (LC040) - Pyramid Pattern

a. Check if level 3 is punched; if not, go to level 4 (LC080).

b. If punched, store level number.

c. Determine tape level to be read (8, 7, or 5).

   d. Space over remaining level control punches.

   e. Generate pyramid pattern.

   f. Read and check one frame.

   g. Repeat from (e) until level 3 is read, then exit to level 4.

6. Level 4 (LC080) - Complement Pyramid Pattern

   a. Check if level 4 is punched; if not, go to level 5 (LC100).

   b. If punched, use the level 3 routine, but read the complement pyramid pattern.

7. Level 5 (LC100) - All Ones, All Zeros Pattern

   a. Check if level 5 is punched; if not, go to level 6 (LC200).

   b. If punched, store level number.

   c. Determine level of tape to be read (8, 7, or 5).

   d. Generate one frame of pattern.

   e. Read and check one frame.

   f. Repeat from (d) until level 5 is read, then exit to next level.

8. Level 6 (LC200) - C9, 36 Pattern

   a. Check if level 6 is punched; if not, check for end of tape punch (LC800).

   b. If punched, store level number.

   c. Determine level of tape to be read (8, 7, or 5).

   d. Generate one frame of pattern.

   e. Read and check one frame.

   f. Repeat from (d) until level 6 is read, then exit to end of tape routine.

9. End of Tape Routine (LC800)

   a. Check for end of tape punch (8-level punch); error stop if not present.

   b. Space over remaining end of tape punches.

   c. Update pass count.

   d. Go to SMM for end of test stop.

   e. Load bias.

   f. Check for repeat test and parameter re-entry.

g. SMM returns control here if frequency number is not zero.

h. Check for re-entry of parameters.

j. Go to (CNTRL) to repeat test.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS

Approximately 850_{10} locations.

B. TIMING - 0 min, 25 sec.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 4K memory.
2. 1721/22/77 per Tape Reader.
3. A device for loading test.

1711/12/13 TELETYPE TEST
(TTY005 Test No. 5)
(CP = 2F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. Sections 4, 5, 6, 7, and 8 are not normally run since they require operator intervention. If these sections are selected, the Teletype Test should be run alone.

2. If the computer stops with overflow light flashing when running sections 5, 6, 7, or 8, the operator must select K mode and press Manual Interrupt to continue.

3. A minimum of four horizontal tabs must be set for section 3 to operate correctly.

4. The paper tapes punched in sections 5 and 7 are used as input in sections 6 and 8 respectively. In the case of a 1713 making mode changes by computer function (parameter MOD ≠ 0), the paper tape must be loaded in the reader before sections 6 or 8 are started.

5. Parameter MOD 1 must be non-zero if teletype is a 1712.

6. Section 6 should be run in both T and KT modes (manual mode selection).

7. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. Call as external test under SMM17.

2. The test may be restarted by MC, set P = IA and RUN.

C. PARAMETERS (in order in which they are entered)

1. SECTNS - Section Select parameter. Prestored as $OEOF_{16}$.

2. COUNTR - set to number of lines to be typed in section 1 and/or number of times bell is to be run in section 11. Prestored as 0040_{16}.

3. MOD - For 1712 operation or a 1713 and manual selection of modes, this parameter must be zero. For 1713 operation with Program mode selection, set to any non-zero value (prestored as zero).

4. MOD 1 - must be non-zero for 1712.

D. MESSAGES

1. Test Title
   TTY005, 1711/12/13 TELETYPE TEST.

2. Initial Address Message
   CP2F, VER. 3.1
   IA = XXXX, FC = XX
   (XXXX = starting address of test, XX = frequency count)

3. Error Messages

   a. All error messages are in the format specified by SMM17.

   b. When a Data Compare error occurs in sections 6 or 8, the actual and expected data are included in the error typeout.

   c. Description of individual error codes.

Error Code	Description
1	Teletype Not Ready
2	Reply on input - reject expected.
3	Internal reject on status request
4	Internal reject on data output
7	Internal reject on function select
8	Improper status (other than Ready, Read Mode, Motor On, and EOP)
9	Data interrupt (reject on data output)
10	No Read Mode status
11	Write mode selected, Read mode status up
12	Data interrupt (reject on data input)
13	Busy status not up after data output
15	Data compare error
20	No interrupt status
21	Non-requested interrupt
22	Interrupt generated by ringing bell
24	Alarm interrupt but no alarm conditions

Error Code	Description
27	Lost data
29	EOP interrupt did not occur
30	Delay adjustment error (adjust delay on bottom even side of Z20-4 card located at C-15 so that controller never goes Not Busy when reading continuously from paper tape).

E. ERROR STOPS

1. FIRST STOP

   (A) = ID word   (Q) = Stop/Jump parameter

2. SECOND STOP

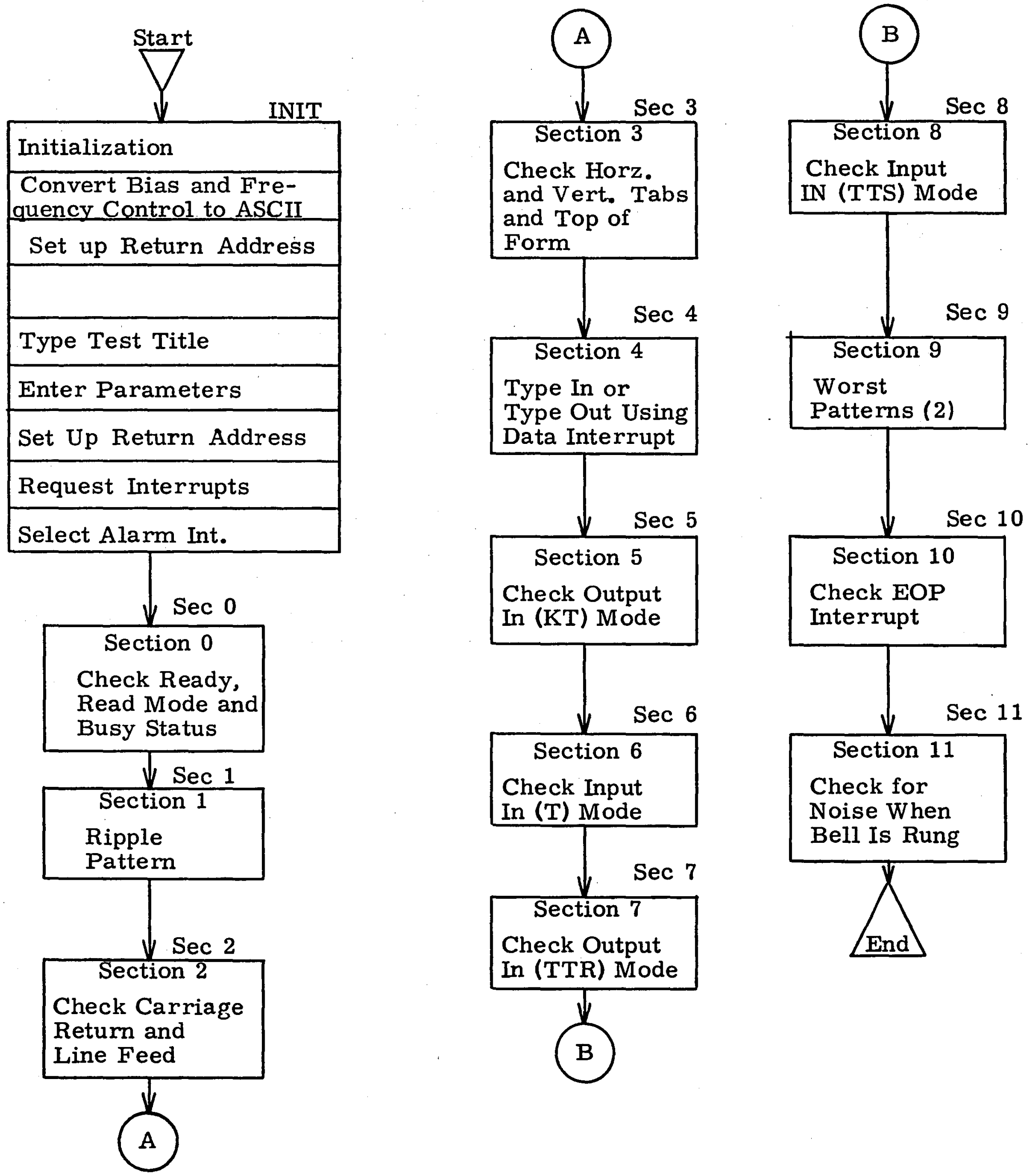
   (A) = section number/ error code   (Q) = return address

3. THIRD STOP (used in Sections 6 and 8)

   (A) = actual data   (Q) = expected data

II. DESCRIPTION

A. BLOCK DIAGRAM



B. SECTION DESCRIPTION

1. Initialization (INIT 0)
   - a. Convert bias address and frequency count to ASCII and store in typeout routine.
   - b. Set up return address.
   - c. Type out test title.
   - d. Return control to SMM.
   - e. Enter parameters if selected.
   - f. Set up return address.
   - g. Request interrupts.
   - h. Select alarm interrupt.
   - j. Return control to SMM.
2. Section 0; Status Check (Sec 0)
   - a. Clear controller.
   - b. Check status. Error if status other than Read Mode, Ready, and End of Operation.
   - c. Check Ready and Read Mode status bits. Error if not present.
   - d. Select Write Mode, check for Read Mode status. Error if present.
   - e. Select Read Mode and check for Read Mode status. Error if not present.
   - f. Repeat from step b 500 times.
   - g. Output a character and check for Busy 10 times. Error if Not Busy.
   - h. Repeat step g six times.
   - j. Check for Repeat Section.
   - k. Go to next section.
3. Section 1; Ripple Pattern (Sec 1)
   - a. Check parameter for number of lines requested and set up line counter.
   - b. Output one line of ASCII code table.
   - c. Shift line left one character and output next line.
   - d. Repeat step c for the requested number of lines.
   - e. Check for Repeat Section.
   - f. Go to next section.

4. Section 2; Check Carriage Return and Line Feed (Sec 2)

   a. Typeout two characters (CR).

   b. Carriage return and line feed.

   c. Space to end of previous typeout.

   d. Repeat from step a until one line is typed.

   e. Check for Repeat Section.

   f. Go to next section.

5. Section 3; Check Horizontal and Vertical Tab and Top of Form Functions (Sec 3)

   a. Output horizontal tab code.

   b. Typeout: HTAB.

   c. Repeat from step a three times.

   d. Carriage return, line feed.

   e. Repeat from step a three times.

   f. Output vertical tab code.

   g. Typeout: VTAB.

   h. Repeat from f three times.

   i. Carriage return, line feed.

   j. Repeat from f.

   k. Output Top of Form code.

   l. Typeout: FORM.*

   m. Repeat from k.

   n. Two carriage return, line feed combinations.

   o. Check for Repeat Section

   p. Go to next section.

6. Section 4; Type Out Random Data Typed In by Operator (Sec 4)

   a. Type out instruction to operator: TYPE IN RANDOM DATA, (200 Character Maximum) PRESS MANUAL INTERRUPT (bell rings twice).

   b. Read in and store data as typed using data interrupt.

   c. Check for Manual Interrupt.

*Typeout may not be on one line since Busy status does not remain up until paper motion is completed.

    d. When Manual Interrupt occurs, type out stored data also using data interrupts.

    e. Check for Repeat Section.

    f. Go to next section.

7. Section 5; Check Output in KT Mode (Sec 5)

    a. Output instruction to operator: SELECT (KT) MODE. PRESS MANUAL INTERRUPT. *

    b. Punch out paper tape leader.

    c. Generate and store 434 random characters with carriage return and line feed placed at appropriate places.

    d. Punch and type out stored data.

    e. Punch out paper tape leader.

    f. Check for Repeat Section.

    g. Go to next section.

8. Section 6; Check Input in T or KT Mode (Sec 6)

    a. Output instruction typed to operator: LOAD TAPE PUNCHED IN SECTION 5, SELECT (T) MODE, RUN PAPER TAPE READER. *

    b. Wait for data.

    c. Read in data from paper tape reader (data typed at same time).

    d. Compare data read into actual data. If Compare error occurs, display actual and expected data during Error stop.

    e. Check for Repeat Section.

    f. Go to next section.

9. Section 7; Check Output in TTR Mode (Sec 7)

    a. Output instruction typed to operator: SELECT (TTR) MODE. PRESS MANUAL INTERRUPT). *

    b. Wait for Manual Interrupt.

    c. Punch out paper tape leader.

    d. Output and store a 217-word block of the Teletype Test from memory.

*Omitted in the case of a 1713 selecting modes by computer function.

e. Check for Repeat Section.

f. Go to next section.

10. Section 8; Check Input in TTS Mode (Sec 8)

a. Output instruction typed to operator: LOAD TAPE PUNCHED IN SECTION 7, SELECT (TTS) MODE, RUN PAPER TAPE READER. *

b. Wait for data.

c. Input and store data from paper tape reader.

d. Compare data read into actual data stored in section 7. If data does not compare, display actual and expected data during Error stop.

e. Check for Repeat Section.

f. Go to next section.

11. Section 9; Worst Patterns (2) (Sec 9)

a. Output five lines of first pattern (*U).

b. Output five lines of second pattern (=F).

c. Check for Repeat Section.

d. Go to next section.

12. Section 10; Check End of Operation Interrupt (Sec 10)

a. Clear interrupts.

b. Select Write mode.

c. Output a character (E), Error stop if internal reject.

d. If reply received, select End of Operation interrupt.

e. Exit to SMM.

f. Check for EOP interrupt occurrence, Error stop if interrupt did not occur.

g. Repeat from step c 50 times.

h. Check for Repeat Section.

j. Go to next section.

13. Section 11; Check for Interrupt Generated by Ringing Bell (Sec 11)

a. Load counter with parameter (COUNTR), 40_{16} unless changed during parameter stop.

*Omitted in the case of a 1713 selecting modes by computer function.

b. Select Alarm interrupt.

c. Ring bell.

d. Check status for interrupt, error if present.

e. Update counter.

f. Repeat from step c until counter goes to zero.

g. Check for Repeat Section.

h. Go to End of Test routine.

14. Interrupt Routine (INT)

a. Control is transferred to location INT when an interrupt occurs.

b. Store (Q), exit interrupt state value.

c. Check status for interrupt bit (error if not set).

d. Check whether section 10 is being executed; if so, go to step s.

e. Check sections 6 or 8 are being executed; if so, go to step h.

f. Check for Alarm interrupt; if not set, go to step h.

g. If Alarm interrupt, check for alarm conditions (Not Ready, Lost Data, or End of Tape*) error if none is present.

h. Check if interrupt was one requested (Data, Manual, or EOT), error if not requested.

j. Check for Manual interrupt; if not present, go to m.

k. Was Manual interrupt requested: If so, set occurrence flag; if not, return control to SMM.

l. Is Section 4 being executed? If not, clear interrupts and exit Interrupt state.

m. Was it a data interrupt? If not, go to step s.

n. Is an output requested? If not, go to step y.

p. Output data, update counter; if counter is zero, clear controller, clear interrupts, and exit Interrupt state.

q. If counter is not zero, exit Interrupt state.

*End of Tape Status is the Alarm Bit set, Ready Bit Set, and Lost Data Bit not set.

r. Input and store data, update counter, clear interrupt status, and exit interrupt State.

s. Check for End of Operation interrupt, set flag. If present, clear interrupts and exit Interrupt state.

t. If EOP is not present, exit Interrupt state.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - approximately 2500_{10} locations.

B. TIMING - 3 min. 45 sec. with 10 lines typed in section 1.

C. EQUIPMENT CONFIGURATION

1. 1704 with 4K memory.
2. 1711/12/13 Teletype
3. A device for loading program if configuration is a 1711.

1740/501, 1742, 1742-20/30 LINE PRINTER TEST
(LP100C Test No. 0C)
(CP = 2F)

I. OPERATING INSTRUCTIONS

A. RESTRICTIONS

1. The three line printer interrupts, Data, End of Operation, and Alarm, may be on separate interrupt lines or on a common line to run this test.
2. Section 10, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to cause an Alarm condition on the line printer, for example, by opening the interlock. He must clear the Alarm condition at the end of the section for further testing.
3. Section 11, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to set the Protect switches on the 1704 console and on the line printer. He must clear the Protect switch on the console at the end of the section for further testing.
4. Section 12, an optional section, assumes that all the format tape levels may be selected sequentially without the paper tearing.
5. Section 13, an optional section, assumes that a change from 6 to 8 lines per inch may be made when 1742 status bit 9 is set. Do not run section 13 on the 1740/501.
6. This test will run on the 1742 and on the 1740. The 1740 does not have 6/8 line per inch select so do not run section 13 on this line printer. The 1740/501 has only eight format function codes while the 1742 has 12 format function codes. The function codes that select format channels 8 through 11 are not used in the 1740/501. There may be a print out for these levels, but they can be ignored.
7. Bits 2 and 3 of the SMM parameter word must specify the correct machine type.

B. LOADING PROCEDURE

1. Call as external test number C under SMM17.
2. Test may be restarted by MC, set P = IA, and RUN.

C. PARAMETERS

If bit 0 of the SMM Stop/Jump parameter is set at the start of the test, or if bits 10 and 0 are set at the start of succeeding passes through the test, a parameter stop occurs.

1. First Stop, A = 0C41, Q = Stop/Jump parameter. The Stop/Jump parameter may be changed if desired.
2. Second Stop, A = 00FE, Q = 0004. The bits in the A register specify the sections to be tested; bit 1 implies section 1, bit 2 implies section 2, etc. The section selection may be changed as desired. The bit in the Q register specifies the Data interrupt line. Interrupt line 0 is internal and must not be chosen for the 1742 or 1740/501.
3. Third Stop, A = 0004, Q = 0004. The bit in the A register specifies the End of Operation interrupt line. The bit in the Q register specifies the Alarm interrupt line. These interrupt line selections must be identical with the physical interrupt line connections. The interrupt line is assumed to be common and is prestored as line 2.
4. Fourth Stop, A = 004D. The A register contains the character code for the character to be used in section 8 (an optional section), the Clarify Test. This character code may be changed as desired to any legal character code.

   Q = 0 = 1742 Q = 1 = 1742-20/30 LINE PRINTER FLAG.

D. MESSAGES

No typeouts occur if bit 8 of the Stop/Jump parameter is set.

1. Test title and initial address typeout:

```
LP100C, 1740/501, 1742 LINE PRINTER TEST
        CP2F, VER. 3.1
        IA = XXXX, FC = XX
```

   XXXX is the initial address of the test.

2. Status bit 7 set at start of test.

   PROTECT STATUS SET

3. Start of Section 10

   CAUSE ALARM ON LINE PRINTER

   This message is also printed on the line printer, preceded and followed by a page eject. This message instructs the operator to cause an Alarm condition on the line printer (for example, by opening the interlock).

4. Start of Section 11

SET PROTECT SWITCHES ON 1704 CONSOLE AND LINE PRINTER

This message is also printed on the line printer, preceded and followed by a page eject. This message instructs the operator to set the Protect switch on the computer console and to set the Protect switch on the printer.

5. End of 1740/501, 1742 test.

A	Q	A	Q
0C24	S/J parameter	Pass Number	Return Address

6. Error Messages

a. All error messages are in the format specified by SMM17.

A	Q	A	Q
0CX8	S/J parameter	0YZZ	Return Address

X = Number of stops (if any) or number of pairs of words typed (if any)
Y = Section number
ZZ = Error code

Additional information is given, depending on the type of error, if X (number of stops) is greater than 2.

b. Description of individual error codes (See below).

E. ERROR STOPS

Stops occur upon errors if bit 3 of the Stop/Jump parameter word is set. At least two stops occur. Additional stops may occur depending on the type of error.

DESCRIPTION OF INDIVIDUAL ERROR CODES

Error Code	Description
01	Equipment address error (operator error). Test must be called again.
02	Parameter error (operator error). Parameters must be selected again.

03	Internal reject of function.
	A = contents of A when reject occurred
	Q = contents of Q when reject occurred

DESCRIPTION OF INDIVIDUAL ERROR CODES (Cont'd)

Error Code   Description

04   External reject of function.

A = contents of A when reject occurred
Q = contents of Q when reject occurred

05   Internal reject of status input

A = 0   Q = contents of Q when reject occurred

06   External reject of status input

A = 0   Q = contents of Q when reject occurred

07   Incorrect status

A = actual status   Q = expected status
(Additional information in Section 2, loop 1A)

A = word count   Q = current word count

08   No data interrupt when expected after selecting data interrupt (Section 1, loop 4 or 5).

A = status   Q = 0

09   No data interrupt when expected after selecting data interrupt and interrupt clear simultanously (Section 1, loop 6).

A = status   Q = 0

0A   No data interrupt when expected after selecting data interrupt and clear printer simultanously (Section 1, loop 7)

A = status   Q = 0

0B   No data interrupt when expected after selecting data interrupt or after selecting all interrupts.

A = status   Q = 0

0C   No alarm interrupt when expected after output of illegal character or after cause alarm on line printer.

A = status   Q = 0

0D   No end of operation interrupt when expected (Section 4, loop 2 and 5).

A = status   Q = 0

0E   No data interrupt when expected (Section 3, loop 5 and Section 4, loops 4 and 5)

A = status   Q = 0

DESCRIPTION OF INDIVIDUAL ERROR CODES (Cont'd)

Error Code	Description
0F	Interrupt but no interrupt status A = status Q = 0
10	Unrequested interrupt A = status Q = 0
11	No external reject of data when expected after exceeding memory capacity by one word (69 words) A = status Q = 0

12	External data reject
	A = contents of A when reject occurred
	Q = contents of Q when reject occurred

13	Internal data reject
	A = contents of A when reject occurred
	Q = contents of Q when reject occurred

14	Wrong word count
	A = status Q = 0
	A = actual word count Q = expected word count

15	Interrupt status failed to clear A = status Q = 0
16	Internal reject when expecting an external reject A = status Q = 0
17	No external reject when expected (clear, space, data, or during print) A = status Q = 0

18	No End of Operation status occurred on buffer data channel after buffered output of 69 or 68 words.
	A = line printer status
	Q = BDC status before termination
	A = current word address
	Q = expected word address

DESCRIPTION OF INDIVIDUAL ERROR CODES (Cont'd)

Error Code   Description

19   End of Operation status occurred on buffer data channel after output of 67 or 68 words, but word count is wrong.

A = line printer status

Q = BDC status before termination

A = current word address

Q = expected word address

1A   Unexpected End of Operation status occurred after buffered output of 69 words.

A = line printer status

Q = BDC status before termination

A = current word address

Q = expected word address

1B   Unexpected external reject after an expected external reject during print.

A = status   Q = 0

1C   Internal reject on attempt to initiate output on buffer data channel

A = first word address minus one of the buffer area

Q = contents of Q when reject occurred

1D   External reject on attempt to initiate output on buffer data channel

A = first word address minus one of the buffer area

Q = contents of Q when reject occurred

1E   Internal reject on attempt to terminate buffer and get current address

A = not applicable

Q = equipment code

1F   External reject on attempt to terminate buffer and get current address

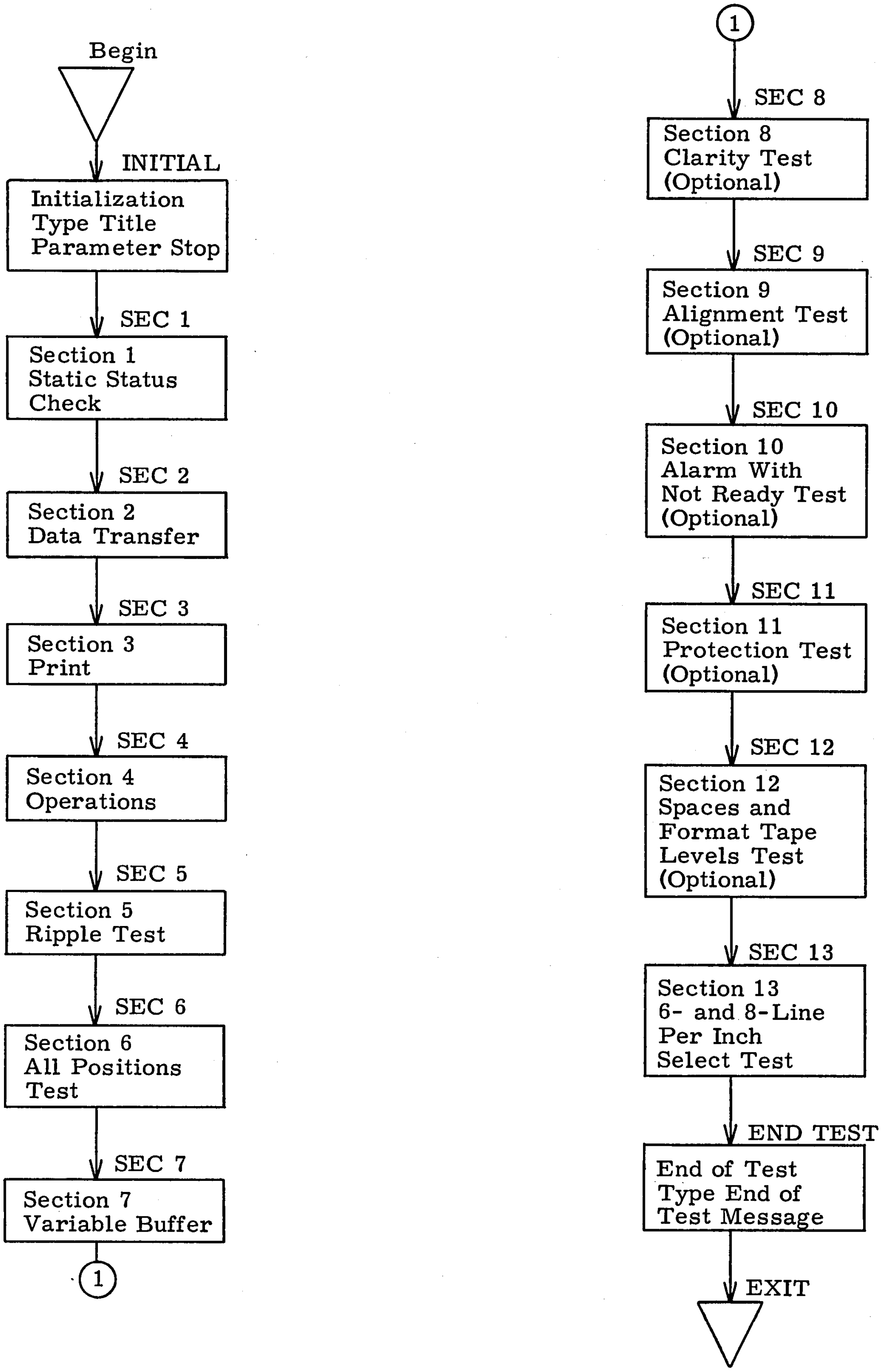
A = not applicable

Q = equipment code

In sections 3 through 9, 12, and 13 Alarm interrupt is selected unless bit 15 of the Stop/Jump parameter is set. If bit 15 of the Stop/Jump is set, the Alarm interrupt selecting is bypassed.

II. TEST DESCRIPTION

A. BLOCK DIAGRAM


Begin
INITIAL
Initialization Type Title Parameter Stop
SEC 1
Section 1 Static Status Check
SEC 2
Section 2 Data Transfer
SEC 3
Section 3 Print
SEC 4
Section 4 Operations
SEC 5
Section 5 Ripple Test
SEC 6
Section 6 All Positions Test
SEC 7
Section 7 Variable Buffer
1
1
SEC 8
Section 8 Clarity Test (Optional)
SEC 9
Section 9 Alignment Test (Optional)
SEC 10
Section 10 Alarm With Not Ready Test (Optional)
SEC 11
Section 11 Protection Test (Optional)
SEC 12
Section 12 Spaces and Format Tape Levels Test (Optional)
SEC 13
Section 13 6- and 8-Line Per Inch Select Test
END TEST
End of Test Type End of Test Message
EXIT

B. TEST DESCRIPTION

0. Initialization

   a. Determine initial and last addresses of test.

   b. Determine whether equipment address is legal.

   c. Type out title.

   d. Get status. Typeout if bit 7 set.

   e. Parameter stop if bit 0 of Stop/Jump word set.

   f. Return control to monitor.

1. Section 1 - STATIC STATUS CHECK

   a. Section 1, Loop 1

      1) Get status, drop bits 7 and 9 if set.
      2) Expect 0019, End of Operation, Data, Ready. Error code 07 if not.
      3) Loop to 1) if bit 4 of Stop/Jump word set.

   b. Section 1, Loop 2

      1) Select Clear Printer.
      2) Get status.
      3) Expect 0019, error code 07 if not.
      4) Loop to 1) if Stop/Jump bit 4 set.

   c. Section 1, Loop 3

      1) Select Clear Interrupts.
      2) Get status.
      3) Expect 0019, error code 07 if not.
      4) Loop to 1) if Stop/Jump bit 4 set.

   d. Section 1, Loop 4

      1) Select Data Interrupt.
      2) Expect interrupt, error code 08 if none.
      3) Get status upon interrupt.
      4) Expect 001D, error code 07 if not.
      5) Clear printer.
      6) Loop to 1) if Stop/Jump bit 4 set.

e. Section 1, Loop 5

Same as Loop 4 except step 5) becomes

5) Select Clear Printer.

f. Section 1, Loop 6

Same as Loop 4 except steps 1) and 2) become

1) Select Data Interrupt and Clear Interrupts simultaneously.
2) Expect interrupt, error code 09 if none.

g. Section 1, Loop 7

Same as Loop 4 except steps 1) and 2) become

1) Select Data Interrupt and Clear Printer simultaneously.
2) Expect interrupt, error code 0A if none.

h. Section 1, Loop 8

1) Select EOP Interrupt.
2) Expect no interrupt, error code 10 if unexpected EOP interrupt.
3) Get status.
4) Expect 0019, error code 07 if not.
5) Clear printer.
6) Loop to 1) if Stop/Jump bit 4 set.

i. Section 1, Loop 9

1) Select Alarm Interrupt.
2) Expect no interrupt, error code 10 if unexpected Alarm interrupt.
3) Get status.
4) Expect 0019, error code 07 if not.
5) Clear printer.
6) Loop to 1) if Stop/Jump bit 4 set.

j. End of Section 1

1) Stop if bit 1 of Stop/Jump word set.
2) Repeat section if bit 5 of Stop/Jump word set.

2. Section 2 - DATA TRANSFER

a. Section 2, Loop 1A

1) Clear printer.
2) Wait Not Busy, get status.

3) Expect 0019, error code 07 if not. Also displayed in error message are the current word count (initially zero) and the current word being outputted (initially 2020, the first character on the drum).
4) Attempt output of a word.
5) Expect no external reject, error code 12 if reject.
6) Get status.
7) Increment word count, loop to 2) if not 68.
8) Wait Not Busy.
9) Get status.
10) Expect 0011, error code 07 if not.
11) Attempt output.
12) Expect external reject, error code 11 if none.
13) Loop to 1) if Stop/Jump bit 4 set.
14) Increment word, loop to 1) if not 6060.

b. Section 2, Loop 1

1) Clear printer and select Data interrupt simultaneously.
2) Expect interrupt, error code 08 if none.
3) In Interrupt routine output a word; do not clear interrupts in Interrupt routine.
4) Wait Not Busy.
5) Expect 68 data interrupts to have occurred, error code 14 if not.
6) Get status.
7) Expect 0011, error code 07 if not.
8) Loop to 1) if Stop/Jump bit 4 is set. Go to section 2, loop 5 if no 1706 (W of the equipment address is zero).

c. Section 2, Loop 2

1) Clear printer.
2) Initiate buffered output of 68 words to line printer via the 1706.
3) Initialize 2-millisecond counter.
4) Expect EOP status on 1706 before counter overflows, error code 18 if none.
5) Get 1706 current word address.
6) Expect equal to contents of first word address minus one, error code 19 if not.
7) Get line printer status.
8) Expect 0011, error code 07 if not.
9) Loop to 1) if Stop/Jump word 4 is set.

d. Section 2, Loop 3

Same as Loop 2, except steps 2) and 8) which are:

2) Initiate buffered output of 67 words to line printer via the 1706.

8) Expect 0019, error code 07 if not.

e. Section 2, Loop 4

1) Clear printer.

2) Initiate buffered output of 69 words to line printer via the 1706.

3) Initialize 2-millisecond counter.

4) Expect no EOP status on 1706 before counter overflows, error code 1A if EOP occurs.

5) Terminate and get 1706 current word address.

6) Expect equal to first word address plus 68, error code 19 if not.

7) Get line printer status.

8) Expect 0011, error code 07 if not.

9) Loop to 1) if Stop/Jump bit 4 is set.

f. Section 2, Loop 5

1) Clear printer.

2) Output illegal character (00).

3) Wait Not Busy, get status.

4) Expect 0039, error code 07 if not.

5) Loop to 1) if Stop/Jump bit 4 is set.

g. Section 2, Loop 6

1) Clear printer.

2) Output illegal character (00).

3) Select Alarm interrupt.

4) Expect interrupt, error code 0C if none.

5) Get status upon interrupt.

6) Expect 003D, error code 07 if not.

7) Clear printer in interrupt routine.

8) Loop to 1) if Stop/Jump bit 4 is set.

h. Section 2, Loop 7

1) Select EOP interrupt, Data interrupt, Clear interrupt, and Clear Printer simultaneously.

2) Expect Data interrupt, error code 0B if none.

3) In Data interrupt routine.

a) Output illegal character (00).

b) Select Alarm interrupt, EOP interrupt, and Clear interrupts simultaneously.

c) Select Clear Printer.

4) Expect Alarm interrupt, error code 0C if none.

5) Get status upon Alarm interrupt.

6) Expect 003D, error code 07 if none.

7) Expect no EOP interrupt, error code 10 if EOP interrupt occurs.

8) Clear printer.

9) Loop to 1) if Stop/Jump bit 4 is set.

j. End of Section 2

1) Stop if bit 1 of Stop/Jump Word is set.

2) Repeat section if Stop/Jump bit 5 is set.

3. Section 3 - PRINT

a. Section 3, Loop 1

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of the Stop/Jump word is clear)

2) Output a word (initially 2021).

3) Increment word, loop to 2) if not 6061.

4) Print (all characters).

5) Get status.

6) Wait Not Busy, get status.

7) Expect 0019, error code 07 if not.

8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

10) Loop to 1) if Stop/Jump bit 4 is set.

b. Section 3, Loop 2

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of of the Stop/Jump word is clear)

2) Output all characters (32 words).

3) Print.

4) Attempt Clear Printer.

5) Attempt Clear interrupt.

6) Expect no external reject, error code 1B if reject occurs.

7) Wait Not Busy.

8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

9) Loop to 1) if Stop/Jump bit 4 is set.

c. Section 3, Loop 3

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of the Stop/Jump word is clear).
2) Output all characters (32 words).
3) Print.
4) Wait Not Busy.
5) Output a word.
6) Expect no external reject, error code 14 if reject occurs.
7) Wait Not Busy.
8) Increment word count (initially zero), loop to 5 if not 68.
9) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
10) Loop to 1) if Stop/Jump bit 4 is set.

d. Section 3, Loop 4

1) Clear printer and select EOP interrupt simultaneously (and select Alarm interrupt simultaneously if bit 15 of the Stop/Jump word is clear).
2) Output all characters (32 words).
3) Print.
4) Wait Not Busy.
5) Expect EOP interrupt, error code 0D if no EOP interrupt.
6) Get status upon interrupt.
7) Expect 001D, error code 07 if not.
8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
9) Loop to 1) if bit 4 of Stop/Jump word is set.

e. Section 3, Loop 5

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of the Stop/Jump word is clear).
2) Output all characters (32 words).
3) Print.
4) Select Data interrupt and Clear interrupts simultaneously (and select Alarm interrupt simultaneously if bit 15 of the Stop/Jump word is clear).
5) Wait Not Busy.
6) Expect Data interrupt, error code 0E if none.
7) Get status upon interrupt.
8) Expect 001D, error code 07 if not.
9) Clear printer in interrupt routine.
10) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
11) Loop to 1) if bit 4 of Stop/Jump word is set.

f. Section 3, Loop 6

1) Clear printer.
2) Output illegal character (00).
3) Wait Not Busy, get status.
4) Expect 0039, error code 07 if not.
5) Print.
6) Wait Not Busy, get status.
7) Expect 0019, error code 07 if not.
8) Clear printer.
9) Loop to 1) if bit 4 of Stop/Jump word is set.

g. End of Section 3

1) Stop if bit 1 of Stop/Jump word is set.
2) Repeat section if bit 5 of Stop/Jump word is set.

4. Section 4 - OPERATIONS

a. Section 4, Loop 1

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump Word is clear).
2) Space.
3) Wait Not Busy, get status.
4) Expect 0019, error code 07 if not.
5) Expect no Alarm interrupt, error code 10, if Alarm interrupt occurs.
6) Loop to 1) if bit 4 of Stop/Jump word is set.

b. Section 4, Loop 2

1) Clear printer and select EOP interrupt simultaneously (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).
2) Space
3) Wait Not Busy.
4) Expect EOP interrupt, error code 0D if no EOP interrupt.
5) Get status upon interrupt.
6) Expect 001D, error code 07 if not.
7) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
8) Loop to 1) if bit 4 of Stop/Jump word is set.

c. Section 4, Loop 3

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear.

2) Space.

3) Attempt Clear Printer.

4) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

5) Loop to 1) if bit 4 of Stop/Jump word is set.

d. Section 4, Loop 4

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).

2) Space.

3) Select Data interrupt.

4) Expect Data interrupt, error code 0E if none.

5) Get status upon interrupt.

6) Expect 000F, error code 07 if not.

7) Select Clear Printer in interrupt routine.

8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

9) Loop to 1) if bit 4 of Stop/Jump word is set.

e. Section 4, Loop 5

1) Clear printer.

2) Space.

3) Select EOP interrupt, Data interrupt, and Clear interrupts simultaneously.

4) Expect Data interrupt, error code 0E if none.

5) Get status upon interrupt.

6) Expect 000F, error code 07 if not.

7) In Data Interrupt routine:

   a) Output illegal character (00).

   b) Select Alarm interrupt, EOP interrupt, and Clear interrupt simultaneously.

8) Expect Alarm interrupt, error code 0C if none.

9) Get status upon interrupt.

10) Expect 002F, error code 07 if not.

11) In Alarm Interrupt routine, select EOP interrupt and Clear interrupt simultaneously.

12) Wait Not Busy.

13) Expect EOP interrupt, error code 0D if not.

14) Get status upon interrupt.

15) Expect 003D, error code 07 if not.

16) Clear printer.

17) Loop to 1) if bit 4 of Stop/Jump word is set.

f. End of Section 4

   1) Stop if bit 1 of Stop/Jump word is set.
   2) Repeat section if bit 5 of Stop/Jump word is set.

5. Section 5 - RIPPLE TEST

   a. Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is set).

   b. Select format level 1 (page eject).

   c. Output all characters in drum order (68 words initially starting with character code 20).

   d. Wait Not Busy.

   e. Print.

   f. Wait Not Busy.

   g. Space.

   h. Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

   i. Loop to c if bit 4 of Stop/Jump word is set.

   j. Shift characters left end-around one character position in computer memory.

   k. Increment line count (initially zero), loop to c if not 136.

   l. Wait Not Busy.

   m. End of Section 5

      1) Stop if bit 1 of Stop/Jump word is set.
      2) Repeat section if bit 5 of Stop/Jump word is set.

6. Section 6 - ALL POSITIONS

   a. Clear printer (and select Alarm interrupt if bit 15 of Stop/Jump word is set).

   b. Output 68 words of one character (initially character code 20).

   c. Wait Not Busy.

   d. Print.

   e. Wait Not Busy.

   f. Space.

   g. Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

h. Loop to b if bit 4 of Stop/Jump word is set.

i. Increment character code, loop to b. if not 60.

j. End of Section 6

   1) Stop if bit 1 of Stop/Jump word is set.
   2) Repeat section if bit 5 of Stop/Jump word is set.

7. Section 7 - VARIABLE BUFFER

   a. Clear printer (and select Alarm interrupt if bit 15 of Stop/Jump word is set).

   b. Section 7, Loop 1 - Ascending Word Count

      1) Output words (initially zero words). Each word (each pair of characters) is equal to the number (in hexadecimal) of words being outputted.
      2) Wait Not Busy.
      3) Print.
      4) Wait Not Busy.
      5) Space.
      6) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
      7) Loop to 1) if Stop/Jump bit 4 is set.
      8) Increment word count, loop to 1) if not 68.

   c. Section 7, Loop 2 - Decrement Word Count

      1) Output words (initially 68 words).
      2) Wait Not Busy.
      3) Print.
      4) Wait Not Busy.
      5) Space.
      6) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
      7) Loop to 1) if bit 4 of Stop/Jump word is set.
      8) Go to Loop 3 if word count is zero.
      9) Decrement word count, loop to 1).

   d. Section 7, Loop 3 - Alternate Ascending and Descending Word Count

      1) Output words (initially zero).
      2) Wait Not Busy.
      3) Print.
      4) Wait Not Busy.
      5) Space.
      6) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

        7) Loop to 1) if bit 4 of Stop/Jump word is set.
        8) Increment line count (initially zero), go to end of section if 138.
        9) If line count is even, word count equals word count (initially zero) for last even line count plus one.
        10) If line count is odd, word count equals word count (initially 68) for last even line count minus one. That is, the word count will be 0, 68, 1, 67, 2, 66, through 68, 0.
        11) Loop to 1) for next line.

    e. End of Section 7

        1) Wait Not Busy.
        2) Stop if bit 1 of Stop/Jump word is set.
        3) Repeat section if bit 5 of Stop/Jump word is set.

8. Section 8 - CLARITY TEST

    a. Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).

    b. Output 68 words of two characters each, the character code being that chosen at the parameter stop, or if no parameter stop, 4D (M).

    c. Wait Not Busy.

    d. Print.

    e. Wait Not Busy.

    f. Space.

    g. Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

    h. Loop to b if bit 4 of Stop/Jump word is set.

    i. Increment line count (initially zero), loop to b if not 120.

    j. End of Section 8

        1) Wait Not Busy.
        2) Stop if bit 1 of Stop/Jump word is set.
        3) Repeat section if bit 5 of Stop/Jump word is set.

9. Section 9 - ALIGNMENT TEST

    Identical to section 8 except the characters printed are hyphens (character code 2D).

10. Section 10 - ALARM WITH NOT READY TEST

a. Preliminary

1) Clear printer.
2) Select format tape level 1 (page eject).
3) Output message.
4) Wait Not Busy.
5) Print message to operator instructing him to cause an Alarm condition on the line printer.
6) Wait Not Busy.
7) Select format tape level 1 (page eject).
8) Output same message on Teletypewriter (if any) preceded and followed by carriage returns.
9) Select Alarm interrupt.
10) Stop with A = Q = 1742.
11) Operator must cause Alarm condition and start computer.

b. Section 10, Loop 1

1) Get status.
2) Expect 003C, error code 07 if not.
3) Loop to 1) if bit 4 of Stop/Jump word is set.

c. Section 10, Loop 2

1) Set bit for Alarm interrupt line in interrupt mask.
2) Expect interrupt, error code 0C if no Alarm interrupt.
3) Get status upon interrupt.
4) Expect 003C, error code 07 if not.
5) Clear bit for Alarm interrupt line in interrupt mask while in interrupt routine.
6) Loop to 1) if bit 4 of Stop/Jump word is set.

d. Section 10, Loop 3

1) Attempt Clear Printer.
2) Loop to 1) if bit 4 of Stop/Jump word is set.

e. Section 10, Loop 4

1) Attempt space.
2) Loop to 1) if bit 4 of Stop/Jump word is set.

f. Section 10, Loop 5

1) Attempt data output.
2) Loop to 1) if bit 4 of Stop/Jump word is set.

g. End of Section 10

1) Stop if bit 1 of Stop/Jump word is set.
2) Operator should clear Alarm conditions on the printer.
3) Repeat section if bit 5 of Stop/Jump word is set.

11. Section 11 - PROTECTION TEST

a. Preliminary

1) Clear printer.
2) Select format tape level 1 (page eject).
3) Output message.
4) Wait Not Busy.
5) Print message to operator instructing him to set the Protect switches on the computer console and on the printer.
6) Wait Not Busy.
7) Select format tape level 1 (page eject).
8) Output same message on Teletypewriter (if any) preceded and followed by carriage returns.
9) Clear Protect bits of all memory locations in computer from address 0000 through last address of test.
10) Stop with A = Q = 1742.
11) Operator must set Protect switches on 1704 and line printer and start computer.

b. Section 11, Loop 1

1) Get status.
2) Expect bit 7 to be set, error code 07 if not.
3) Loop to 1) if bit 4 of Stop/Jump word is set.

c. Section 11, Loop 2

1) Attempt Clear Printer.
2) Expect external reject, error code 17 if not.
3) Loop to 1) if bit 4 of Stop/Jump word is set.

d. Section 11, Loop 3

   1) Attempt space.
   2) Expect external reject, error code 17 if not.
   3) Loop to 1) if bit 4 of Stop/Jump word is set.

e. Section 11, Loop 4

   1) Attempt data output.
   2) Expect external reject, error code 17 if not.
   3) Loop to 1) if bit 4 of Stop/Jump word is set.

f. End of Section 11

   1) Stop if bit 1 of Stop/Jump word is set.
   2) Operator must clear Protect switch on the computer.
   3) Repeat section if bit 5 of Stop/Jump word is set.

12. Section 12 - SPACES AND FORMAT TAPE LEVELS TEST

a. Section 12, Loop 1

   1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).
   2) Print out words "SINGLE SPACE".
   3) Wait Not Busy.
   4) Print.
   5) Wait Not Busy.
   6) Single space.
   7) Increment line count (initially zero), loop to 2) if not 9.
   8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.
   9) Loop to 2) if bit 4 of Stop/Jump word is set.

b. Section 12, Loop 2

   Same as Loop 1 except for steps 2) and 6) which become:

   2) Print out words "DOUBLE SPACE".
   6) Double space.

c. Section 12, Loop 3

   1) Clear printer (and select Alarm interrupt if bit 15 of the Stop/Jump word is clear).
   2) Select format tape level (initially level 1).

3) Print out words: FORMAT TAPE LEVEL 0X, where X is the format tape level in hexadecimal. Also print out words "TOP OF FORM", if level 1, or "LAST LINE OF FORM" if level 12, or "0Y SPACE (s) SHOULD PRECEDE THIS LINE" if level 2 through 11, where Y = 1 for levels 2 and 7, Y = 2 for levels 3 and 8, Y = 3 for levels 4 and 9, Y = 4 for levels 5 and 10, and Y = 5 for levels 6 and 11. (For the 1740/501, Y = 1 for levels 2 and 7, Y = 2 for level 3, Y = 3 for level 4, Y = 4 for level 5, and Y = 5 for level 6.)

4) Wait Not Busy.

5) Print.

6) Wait Not Busy.

7) Loop to 2) if bit 4 of Stop/Jump word is set.

8) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

9) If level equals 12, go to end of section.

10) Increment level, loop to 2.

d. End of Section 12

1) Clear printer.

2) Stop if bit 1 of Stop/Jump word is set.

3) Repeat section if bit 5 of Stop/Jump word is set.

13. Section 13 - 6 AND 8 LINE SELECT TEST

a. Section 13, Loop 1

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).

2) Select format tape level 1 (page eject).

3) Wait Not Busy, get status.

4) Expect status bit 9, error code 07 if not set.

5) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

6) Loop to 1) if bit 4 of Stop/Jump word is set.

b. Section 13, Loop 2

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).

2) Space.

3) Wait Not Busy.

4) Increment line count (initially zero), skip to step 8) if 4.

5) Get status.

6) Expect no bit 9, error code 07 if set.

7) Loop to 2).

8) Get status.

9) Expect bit 9, error code 07 if not set.

10) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

11) Loop to 1) if bit 4 of Stop/Jump word is set.

c. Section 13, Loop 3

1) Clear printer (and select Alarm interrupt simultaneously if bit 15 of Stop/Jump word is clear).

2) Select eight lines per inch.

3) Print out wores "08 LINES PER INCH".

4) Wait Not Busy.

5) Print.

6) Wait Not Busy.

7) Space.

8) Increment line count (initially zero), loop to 3) if not 16.

9) Expect no Alarm interrupt, error code 10 if Alarm interrupt occurs.

10) Loop to 1) if bit 4 of Stop/Jump word is set.

d. Section 13, Loop 4

Same as Loop 3 except step 2) is omitted and step 3) becomes:

3) Print out words "06 LINES PER INCH".

e. End of Section 13

1) Stop if bit 1 of Stop/Jump word is set.

2) Repeat section if bit 5 of Stop/Jump word is set.

14. End of 1740/501, 1742 Test

a. Type out end of test message if bit 8 of Stop/Jump word is clear.

b. Stop if bit 2 of Stop/Jump word is set.

c. Repeat test if bit 6 of Stop/Jump word is set, exit test if not. Monitor may run test again.

d. If test is repeated, parameter stop occurs if bits 10 and 0 (and 6) are set in Stop/Jump word.

III. PHYSICAL REQUIREMENTS

A. SPACE REQUIRED - about 2500_{10} locations

B. TIMING - 1 min. 30 sec.

C. EQUIPMENT CONFIGURATION

1. 1704 Computer with 4K memory
2. 1705 Interrupt Data Channel
3. 1706 Buffer Data Channel (optional)
4. 1742 Line Printer or a 1740 Line Printer Controller and a 501 Line Printer
5. A device for loading program

The following special characters will differ on the pattern print out depending on whether the BCD or ASCII drum is used.

	DRUM SYMBOL	
	BCD	ASCII
Not equal	≠	
(Apostrophe)		'
Less than or equal	≤	
(Ampersand)		&
Arrow Right	→	
(At Sign)		@
Identity	≡	
(Underline)		_
Logical and	∧	
(Question Mark)		?
Logical or	∨	
(Exclamation point)		!
Logical not	¬	
(Quotation Mark)		"
Arrow Up	↑	
(Number Sign)		#
Arrow Down	↓	
(Reversed slash)		/
Greater than or equal	≥	
(Circumflex)		^

FF524-A/1742-120/512 PRINTER TEST
(LP5A23 Test No. 23)

I. OPERATING PROCEDURE

A. RESTRICTIONS

1. This test must be run alone.
2. This test assumes a format tape has already been installed and punched as shown in Table 1.

B. LOADING PROCEDURE

1. This test is called through normal procedure from SMM17.
2. Parameter Stops

First Stop:	A = 2341	Q = Stop/Jump Word
Second Stop:	A = Section Select Bit - Bit 0 corresponds to Section 0. Bit 1 corresponds to Section 1, etc.	

Q = Train Select Set Q = 0000 for 63 character train. Set Q = 8000 for 48AN Train Set Q = Any positive number for 48HN Train.

Third Stop:	A = Data Interrupt Line	Q = End of Operation Interrupt Line
Fourth Stop:	A = Alarm Interrupt	Q = N.A.

Section Select Assignments

Bit	Section	Title
0	0	Status Check
1	1	Spacing Test
2	2	Interrupt Status
3	3	Ripple Left Test
4	4	Ripple Right Test
5	5	Hammer Clarity
6	6	On Character in Alternating Position
7	7	Variable Buffer
8	8	Buffer Memory Test
9	9	Format Level Test
10	A	Image Memory Test
11	B	Random Pattern Test

II. MESSAGES

A. NORMAL MESSAGES

1. Title of Test: LP5A23 1742-120 LINE PRINTER W/595-4 TRAIN. CP2C
   IA=XXXX XXXX = INITIAL ADDRESS OF TEST.

TABLE 1. FORMAT TAPE CONFIGURATION TO BE USED WITH 1742-120 TESTS

Frame	Levels to be Punched												Frame	Levels to be Punched											
	1	2	3	4	5	6	7	8	9	10	11	12		1	2	3	4	5	6	7	8	9	10	11	12
1 - 17	X	X	X	X	X	X	X	X	X	X	X		34 - 100			X						X		X	
2 - 18											X		35 - 101		X									X	
3 - 19		X									X		36 - 102					X						X	
4 - 70			X						X		X		37 - 103		X	X	X		X			X		X	
5 - 71		X		X							X		38 - 104											X	
6 - 72					X						X		39 - 105		X									X	
7 - 73		X	X			X			X		X		40 - 106			X						X		X	
8 - 74							X				X		41 - 107		X		X	X			X		X	X	
9 - 75		X		X				X			X		42 - 108											X	
10- 76			X						X		X		43 - 109		X	X						X		X	
11- 77		X			X					X	X		44 - 110											X	
12- 78											X		45 - 111		X		X							X	
13- 79		X	X	X		X			X		X		46 - 112			X		X				X		X	
14- 80											X		47 - 113		X									X	
15- 81		X					X				X		48 - 114											X	
16- 82			X		X				X		X		49 - 115		X	X	X		X		X	X		X	
17- 83		X		X				X			X		50 - 116							X				X	
18- 84											X		51 - 117		X			X					X	X	
19- 85		X	X			X			X		X		52 - 118			X						X		X	
20- 86											X		53 - 119		X		X							X	
21- 87		X		X	X					X	X		54 - 120											X	
22- 88			X				X		X		X		55 - 121		X	X			X			X		X	
23- 89		X									X		56 - 122					X						X	
24- 90											X		57 - 123		X		X			X	X			X	
25- 91		X	X	X		X		X	X		X		58 - 124			X						X		X	
26- 92				X							X		59 - 125		X									X	
27- 93		X									X		60 - 126											X	
28- 94			X						X		X		61 - 127		X	X	X	X	X			X	X	X	
29- 95		X		X		X					X		62 - 128											X	
30- 96											X		63 - 129		X									X	
31- 97		X	X		X	X			X	X	X		64 - 130			X				X		X		X	X
32- 98											X		65 - 131												
33- 99		X		X				X			X		66 - 132												

Cut the tape on the line at Frame *132 and glue together. After the tape is glued into a loop be sure to re-punch the holes in the last two frames.

B. ERROR STOPS

1. All error stops are set up like normal SMM17 errors.

First Stop: A = Ident Word
Q = Stop /Jump Word

Second Stop: A = XYWW
X = Section
Y = Subsection
WW = Error Code
Q = Return Address (Biased)

If there are more than two stops, an explanation will be given in the description of the error.

Error Code	Description
01	Printer not ready
02	False busy status
03	No busy status when there should be
07	No data interrupt
08	No End of Operation interrupt. The third stop will be: A = status Q = 0000
0B	Unexpected interrupt. A third stop will show: A = status Q = 0000
12	No alarm interrupt. A third stop will show: A = status Q = 0000
1D	Busy too long. If busy status did not drop within 750 milliseconds after a print operation, paper motion, or error memory read, this error is given

Error Code	Description
1E	External reject on output. A third stop will show: A = status Q = 0000
1F	External reject on input
20	Internal reject on output. A third stop will show: A = status Q = 0000
21	Internal reject on input
25	Interrupt bit not set on interrupt. A third stop will show: A = status Q = 0000
26	Data bit not set in status. A third stop will show: A = status Q = 0000
27	End of operation bit not set on EOP. A third stop will show: A = status Q = 0000
28	Alarm bit not set on alarm interrupt. A third stop will show: A = status Q = 0000
29	Abnormal End of Operation bit not set on alarm interrupt. A third stop will show: A = status Q = 0000

III. SECTION DESCRIPTIONS

A. SECTION 0

1. Ready Status (Subsection 0)

   a. Perform a clear printer function.

   b. Check for ready status.

   c. Repeat steps a through b 50 times. If not ready error 1 (printer not ready).

2. Busy Status - During No Operation (Subsection 1)

   a. Perform a clear printer.

   b. Check for busy status. If busy, error 2 (false busy status).

   c. Repeat steps a through b 50 times.

3. Busy Status - During and After Print Operation (Subsection 2)

   a. Perform a clear printer.

   b. Check for busy status. If busy, error 2 (false busy status).

   c. Print "During and after the printing of this data status responses are being checked".

   d. Check busy status. If not busy, error 3 (no busy status when there should be).

   e. Wait for busy to drop. If not dropped by 750 milliseconds, error 1D (busy too long).

   f. Repeat steps c through e 10 times.

B. SECTION 1

1. Single Space Six Lines Per Inch (Subsection 0)

   a. Advance to top of form and print: "Single space six lines per inch".

   b. Single space, suppress space, print: "Function Code 01 is for single spacing".

   c. Repeat step b 25 times.

2. Double Space Six Lines Per Inch (Subsection 1)

   a. Advance to top of form and print: "Double space six lines per inch".

   b. Double space.

   c. Suppress space, print: "Function code 02 is for double spacing", double space.

   d. Repeat step c 25 times.

3. Last Line of Form (Subsection 2)

   a. Advance to last line of form. Print "Last line of form".

   b. Repeat step a three times.

4. Top of Form (Subsection 3)

   a. Function last line of form.

   b. Print "Last line of form".

   c. Advance to top of form. Suppress space and print "top of form".

   d. Repeat c two times, each time addressing another TOF message. The first TOF message will be printed over itself three times, the second one two times, and the third one once.

5. Page Eject (Subsection 4)

   a. Function page eject.

   b. Print "Top of form".

   c. Repeat a through b three times.

C. SECTION 2

1. Interrupt on Data Function (Subsection 0)

   a. Perform a clear printer.

   b. Start printing "Checking Data Interrupt Status".

   c. Function Data interrupt.

   d. Wait for busy status to drop.

   e. Verify Data interrupt. If no interrupt, error 7 (no Data interrupt).

   f. Check interrupt bit. If not set, error 25 (interrupt bit not set on interrupt).

   g. Check data bit. If not set, error 26 (data bit not set on data interrupt).

   h. Repeat steps b through g 25 times.

2. Check Clear Interrupt on Data Function

   a. Perform a clear printer.

   b. Start printing "Checking Clear Data Interrupt Function".

   c. Function Data interrupt.

   d. Function Clear Data interrupt.

   e. Wait for busy status to drop. If interrupt, error B (unexpected, interrupt).

   f. Repeat steps b through e 10 times.

3. Interrupt on End of Operation - Print Operation (Subsection 2)

   a. Perform a clear printer.

   b. Start printing "Checking End of Operation Interrupt-Print Operation".

   c. Function EOP interrupt.

   d. Wait for busy status to drop.

   e. Verify EOP interrupt. If no interrupt, error 08 (no EOP interrupt).

f. Check Interrupt bit. If not set, error 25 (interrupt bit not set on interrupt).

g. Check data bit. If not set, error 26 (data bit not set on data ready).

h. Check EOP bit. If not set, error 27 (end of operation bit not set on end of operation).

i. Repeat steps b through h 10 times.

4. Interrupt on End of Operation - Paper Motion (Subsection 3)

   a. Same as subsection 2 with the exception of step b. Step b is replaced with: Start printing "Checking End of Operation Interrupt - Paper Motion".

      Wait for busy status to drop.

      Function a double space.

      This test is repeated 10 times.

5. Check Clear End of Operation Interrupt (Subsection 4)

   a. Function EOP interrupt.

   b. Function clear EOP interrupt.

   c. Start printing "Checking Clear End of Operation Interrupt Function".

   d. Wait for busy status to drop.

   e. If interrupt is received, error 0B (unexpected interrupt).

   f. Repeat steps a through e 10 times.

6. Check Alarm Interrupt

   a. Function Alarm interrupt.

   b. Output 10 codes.

   c. Function print.

   d. Wait for busy to drop.

   e. Verify interrupt. No interrupt, error 12 (no alarm interrupt).

   f. Check Interrupt bit. If not set, error 25 (interrupt bit not set on interrupt).

   g. Check alarm bit. If not set, error 28 (alarm bit not set on alarm interrupt).

   h. Check abnormal EOP bit. If not set, error 29 (abnormal EOP bit not set on alarm interrupt).

   i. Repeat steps a through h 10 times.

7. Check Clear Alarm Interrupt

   a. Function clear printer.

   b. Function alarm interrupt.

   c. Function clear alarm interrupt.

   d. Output 10 codes.

   e. Function print.

   f. Verify no interrupt. If interrupt occurs, error 0B (unexpected interrupt).

D. SECTION 3

1. Ripple Left - six lines per inch

   a. Print one line of all characters on train.

   b. Shift line one character left for each line of print.

   c. Print 136 lines.

2. Ripple Left - eight lines per inch

   a. Same as subsection 1 at only eight lines per inch.

E. SECTION 4

1. Ripple Right - six lines per inch

   a. Print one line of all characters on train.

   b. Shift line one character left for each line of print.

   c. Print 136 lines.

F. SECTION 5

1. Hammer Adjustment and Clarity - six lines per inch

   a. Print 14 lines of the letter H in even columns.

   b. Print 14 lines of the letter H in odd columns.

   c. Alternate rows of M and W are printed, 28 lines.

G. SECTION 6

1. Print One Character Alternating with a Space - six lines per inch

   a. Print five lines of a character alternating with a space.

   b. Single space between each five line group.

   c. Repeat a and b for each character on train.

H. SECTION 7

1. Variable Buffer - six lines per inch

   a. Print one line of one word (two characters).

   b. Increase each successive line by one word until one full line is printed.

   c. Decrease each successive line by one word until one word is left.

I. SECTION 8

1. Buffer Memory Test - six lines per inch

   a. Change image code for 4 to $CC.

   b. Print one full line of 3 and 4.

   c. Repeat a and b 32 times.

   d. Print one full line of 4 and 3.

   e. Repeat d 32 times.

J. SECTION 9

1. Format Tape Level Test (pre-print levels)

   a. All pre-print levels are selected in such an order as to show a double space between each printed line. Each line that is printed gives the pre-print level that was selected. All lines are double spaced including the last line, level 12. The following is the order in which the preprint levels were selected.

1
4
6
8
10
7
6
9
10
11
6

4
5
4
9
8
6
4
11
4
7
8

6
4
5
7
6
8
9
4
5
12

K. SECTION A

1. Image Memory Test

a. The first 48 locations are filled with a parity pattern (Figure 1). The remaining locations are loaded with a $33 code.

b. The first 48 locations of the buffer memory are filled with the same pattern. The remaining locations are space codes.

c. Print one line.

d. Shift parity pattern one location to the right. A $33 code replaces the location just shifted.

e. Repeat c through d until pattern has been shifted through all 288 locations. Double space between every 10 lines.

Figure 1. Parity Pattern For Section A

1.	01	13.	07	25.	0D	37.	13
2.	FE	14.	F8	26.	F2	38.	EC
3.	02	15.	08	27.	0E	39.	14
4.	FD	16.	F7	28.	F1	40.	EB
5.	03	17.	09	29.	0F	41.	15
6.	FC	18.	F6	30.	F0	42.	EA
7.	04	19.	0A	31.	10	43.	16
8.	FB	20.	F5	32.	EF	44.	E9
9.	05	21.	0B	33.	11	45.	17
10.	FA	22.	F4	34.	EE	46.	E8
11.	06	23.	0C	35.	12	47.	18
12.	F9	24.	F3	36.	ED	48.	E7

L. SECTION B

1. Random Pattern Test

a. Image memory is reloaded with codes 0 through 3E.

b. One full line of random characters is generated.

c. Step b is repeated 132 times. If a print error occurs, an alarm interrupt will be received.
An error 0B (unexpected interrupt) will be received.

	Director Functions
A00=1	Clear Printer
A01=1	Clear Interrupt
A02=1	Data Interrupt Request
A03=1	End of Operation (EOP) Interrupt Request
A04=1	Alarm Interrupt Request
A05=1	Print Request
A06-A15	Not Used

	Status Codes
XXX1	Ready
XXX2	Busy
XXX4	Interrupt
XXX8	Data
XX1X	End of Operation
XX2X	Alarm
XX4X	Abnormal End of Operation
XX8X	Protected
X1XX	Load Image

ASCII CODES FOR EACH CHARACTER ON THE TRAIN

Codes	63	AN	HN
20	Space	Space	Space
21	∪	None	None
22	¬	+	&
23	↑	None	None
24	$	$	$
25	%	None	⌐
26	≤	None	None
27	#	@	None
28	(	%	(
29	)	*	)
2A	*	*	*
2B	+	&	+
2C	, (Comma)	,	,
2D	-	-	-
2E	.	.	.
2F	/	/	/
30	0	0	0
31	1	1	1
32	2	2	2
33	3	3	3
34	4	4	4
35	5	5	5
36	6	6	6
37	7	7	7
38	8	8	8
39	9	9	9
3A	:	None	None
3B	;	None	None
3C	< (Less)	None	None
3D	=	*	=
3E	> (Greater)	None	None
3F	≥	None	None
40	↱	None	None
41	A	A	A
42	B	B	B
43	C	C	C

Codes	63	AN	HN
44	D	D	D
45	E	E	E
46	F	F	F
47	G	G	G
48	H	H	H
49	I	I	I
4A	J	J	J
4B	K	K	K
4C	L	L	L
4D	M	M	M
4E	N	N	N
4F	O	O	O
50	P	P	P
51	Q	Q	Q
52	R	R	R
53	S	S	S
54	T	T	T
55	U	U	U
56	V	V	V
57	W	W	W
58	X	X	X
59	Y	Y	Y
5A	Z	Z	Z
5B	(	None	None
5C	↓	None	None
5D	)	None	None
5E	∧	None	None
5F	≡	None	None

	Train Set							
Order	63-1	63-2	63-3	63-4	63-5	63-6	AM	HM
1	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1
3	2	2	2	2	2	2	2	2
4	3	3	3	3	3	3	3	3
5	4	4	4	4	4	4	4	4
6	5	5	5	5	5	5	5	5
7	6	6	6	6	6	6	6	6
8	7	7	7	7	7	7	7	7
9	B	$	J	V	$	(	8	8
10	Z	→	K	W	→	G	9	9
11	,	≠	Q	X	≠	H	G	G
12	/	=	*	Y	=	)	H	H
13	8	8	8	8	8	8	+	+
14	9	9	9	9	9	9	-	-
15	A	A	A	A	A	A	.	.
16	C	C	C	C	C	C	E	E
17	(	B	≤	J	V	≤	A	A
18	G	Z	≥	K	W	≥	B	B
19	H	,	<	Q	X	<	C	C
20	)	/	>	*	Y	>	D	D
21	R	R	R	R	R	R	R	R
22	I	I	I	I	I	I	I	I
23	F	F	F	F	F	F	F	F
24	L	L	L	L	L	L	L	L
25	M	M	M	M	M	M	M	M
26	N	N	N	N	N	N	N	N
27	O	O	O	O	O	O	O	O
28	P	P	P	P	P	P	P	P
29	(	(	B	(	J	V	J	J
30	5	6	7	5	K	W	K	K
31	┐	H	,	┐	Q	X	Q	Q
32	]	)	/	)	x	Q	x	x
33	D	D	D	D	D	D	/	/
34	U	U	U	U	U	U	S	S
35	S	S	S	S	S	S	T	T
36	T	T	T	T	T	T	U	U

	Train Set							
Order	63-1	63-2	63-3	63-4	63-5	63-6	AM	HM
37	V	↑	(	B	↑	J	V	V
38	W	↓	G	Z	↓	K	W	W
39	X	∧	H	,	∧	Q	X	X
40	Y	∨	)	/	∨	x	Y	Y
41	+		+	+	+	+	$	$
42	-	-	-	-	-	-	&	&
43	-	-	-	-	-	-	,	,
44	E	E	E	E	E	E	Z	Z
45	J	V	=	(	B	=		(
46	K	W	;	G	Z	;	#	=
47	Q	X	≡	H	;	≡	@	=
48	*	Y	=	)	/	=	%	)

1729 CARD READER TEST
(CR1006 Test No. 6)
(CP = 01)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. The first ten cards should have alternate columns of 2525 and 5252 through column 78. The remaining cards may be random. All cards must contain Even parity in column 79 and a count of holes (through column 79) in column 80.
2. The test ends when the read station becomes empty or an End of File card is read in section 4.

B. LOADING PROCEDURE

1. The test operates under control of 1700 System Maintenance Monitor (SMM17).
2. The calling sequence is that specified by SMM17.
3. Restart test after loading by MC, set P = IA and RUN.

C. PARAMETERS

1. Normal operation requires no parameters.
2. Selected parameters will be typed out.
3. To alter parameters use the procedure explained in SMM17 Description. The identification word (A) and Stop/Jump parameter (Q) are displayed first. Next parameters are sections to be run in the A register.

   Bit 0 = Section 1
   Bit 1 = Section 2
   Bit 2 = Section 3
   Bit 3 = Section 4

   (Q) = Parameter DEL; this parameter, if set, causes the test to delay the specified number of milliseconds between reading each card in section 4. Preset to zero (no delay).

The next parameters are for a random delay between cards.

(A) = Parameter MAX; the maximum delay between cards (in milliseconds). Prestored as zero (no delay).

(Q) = Parameter MIN; the minimum delay between cards (in milliseconds). Prestored as 40_{16}.

Note: For a random delay, the fixed delay parameter (DEL) must be zero and the maximum delay parameter (MAX) must be non-zero.

D. SELECTIVE STOP AND JUMP SETTINGS

Set bit 11 of the Stop/Jump parameter to decrease the number of typeouts for data errors on a 2525 card (see error code B). See bit 4 of the Stop/Jump parameter to repeat conditions which caused certain errors.

E. MESSAGES

1. Normal Program Typeouts
   a. Card Reader Test identification at start of test:

```
CR1006, 1729 CARD READER TEST
IA = XXXX, FC = XX, CP01, VER. 3.1
```

(XXXX = starting address of test, XX = frequency count.)

   b. Typeout if test input was 1729 Card Reader:

```
LOAD 1729 AND PRESS MANUAL INTERRUPT TO START TEST
```

   c. End of 1729 test:

A	Q	A	Q
0624	S/J	PASS NO.	RETURN ADDRESS

2. Error Alarms

   a. identification word

   b. Stop/Jump parameter

   c. section/error code

   d. return address

   e. information (depending on the error code)

   f. information (depending on the error code)

3. Error Codes

Information depending on the error code is typed out and/or displayed in the A and Q registers.

Error Code 1 - Card reader non-existent
A = status
Q = 0

Error Code 3 - Card reader Not Ready
A = status
Q = 0

Error Code 4 - Reader empty (but Ready is set)
A = status
Q = 0

Error Code 5 - Busy not set after start motion
A = status
Q = 0

Error Code 7 - Internal reject of data input
A = status
Q = 0

Error Code 9 - Lost data
A = status
Q = 0

Error Code B - Data error in columns 1-78 of 2525 card.
A = data read
Q = column no.

NOTE

Set bit 11 in the Stop/Jump parameter to ignore the rest of the data errors through column 78 of this card.

Error Code D - Column 79 (parity) of 2525 card is not all one's.
A = data read
Q = 4F (hexadecimal for 79)

Error Code F - Column 80 (hole count) of 2525 card is not 480.
A = data read
Q = 50 (hexadecimal for 80)

Error Code 11 - End of Record status not set after 80th column input.
A = status
Q = 0

Error Code 13 - Lost Data status not set after delaying 20 ms.
A = status
Q = 0

Error Code 15 - Alarm status not set after forcing Lost Data.
A = status
Q = 0

Error Code 17 - External reject of data input after waiting for Data Ready
A = status
Q = 0

Error Code 19 - No interrupt after forcing Lost Data.
A = status
Q = 0

Error Code 1B - No End of Record interrupt after input of 80th column
A = status
Q = 0

Error Code 1D - End of Record status not set when End of Record interrupt occurred
A = status after interrupt
Q = 0

Error Code 1F - End of Record interrupt occurred before input of 80th column
A = status after interrupt
Q = column no.

Error Code 22 - Internal reject of function
A = status
Q = function

Error Code 24 - External reject of function
A = status
Q = function

Error Code 26 - Internal reject of status
A = 0
Q = 0

Error Code 28 - External reject of status
A = 0
Q = 0

Error Code 2A - No interrupt when waiting for Data Ready
A = status
Q = 0

Error Code 2C - Column 80 (hole count) of random card in error
A = data read
Q = expected count

Error Code 2E - Column 79 (parity) of random card in error
A = rows in error
Q = 0

Error Code 32 - Alarm status not set after interrupt
A = status
Q = 0

Error Code 34 - Neither Lost Data nor Reader Empty status set after Alarm interrupt
A = status after interrupt
Q = 0

Error Code 36 - Internal reject of input after Data Ready interrupt
A = status before reject
Q = status after reject

Error Code 38 - External reject of input after Data Ready interrupt
A = status before reject
Q = status after reject

Error Code 3C - Interrupt bit not set
A = status
Q = 0

If bit 4 of the Stop/Jump parameter is set, conditions which caused the error will be repeated for error codes 1, 5, 7, 13, 15, 17, 19, 1B, 1F, 36, 38. The test will loop on errors 3, 4, 22, 24, 26, and 28 until corrected.

4. Error Stops

Stops will occur upon errors if bit 3 in the Stop/Jump parameter is set.

II. DESCRIPTION

Section 1 - Read five 2525 cards, looping on external reject of input. Check for End of Record at end of each card. Check data after each card.

Section 2 - Same as section 1 except wait for Data Ready before input.

Section 3 - First card. Force Lost Data and Alarm status by delaying 20 ms after first Data Ready status appears.

Second card. Force Alarm interrupt to occur by delaying 20 ms after first Data Ready status appears.

Third card. Interrupt on End of Record is selected. Read each column after waiting for Data Ready.

Section 4 - Interrupt on Data Ready and Interrupt on Alarm are selected. Check for End of Record at end of each card. Check hole count of each card. If hole count is in error, check parity. Cards are read until read station becomes empty or an End of File card (7-8 punches in column 1 only) is read.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - approximately 1250 locations.

B. TIMING - variable depending upon the size of the test deck.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 4K memory.
2. 1729 Card Reader.
3. A device for loading the program.

1728 CARD READER/PUNCH TEST

(CRPAOD Test No. D)
(CP=OF)

I. OPERATING INSTRUCTIONS

A. RESTRICTIONS

1. Section 12, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to set the Protect switches on the 1704 console and on the 1728. He must clear the PROTECT switch on the console at the end of the section for further testing.
2. Section 13, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to cause an alarm condition on the 1728, for example, by making the input hopper empty. He must clear the alarm condition at the end of the section for further testing.
3. Sections 7 and 11 will not be executed unless the 1728 is on a 1706 (or 1716) Buffered Data Channel. However, if section 7 is selected, sectior 6 will be executed using a random pattern and a sync check pattern.
4. The cards punched in sections 4 through 7 are to be read in sections 8 through 11. In each of sections 4 through 7, 100 cards are punched followed by an end-of-file card which is offset. More cards will be punched if Stop/Jump bit 4 is set. Cards having punch errors will also be offset.
5. The test can not be loaded into a 4K computer.
6. This test must be run alone.
7. Bits 2 and 3 of SMM parameter word must specify correct machine type.

B. LOADING PROCEDURE

Call as external test number D under SMM17. The equipment address must have bit 0 set and bits 1 through 6 all clear.

Restart test after loading by MC, set P = IA and RUN.

C. PARAMETERS

If bit 0 of the SMM stop/jump word is set at the start of the test or if bits 10 and 0 are set at the start of succeeding passes through the test, a parameter stop occurs. (Selected parameters are typed out.)

1. First stop, A = 0D31, Q Stop/Jump word.

2. Second stop, A = 00FF, Q = 0XYZ. The bits in the A register specify the sections to be tested, i.e., sections 0 through 7. The sections available are:

   Section 0 - Punch Static Check

   Section 1 - Reader Static Check

   Section 2 - Punch Feed, Interrupt Check

   Section 3 - Reader Feed, Interrupt Check

   Section 4 - Punch when No Reject (52 Pattern)

   Section 5 - Punch when Data Status (Shifting Ones Pattern)

   Section 6 - Punch when Data Interrupt (Shifting Zeros Pattern)

   Section 7 - Buffered Output (Random and Single Column Patterns)

   Section 8 - Read when No Reject

   Section 9 - Read When Data Status

   Section 10 - Read when Data Interrupt

   Section 11 - Buffered Read

   Section 12 - Protection Test (optional)

   Section 13 - Alarm with Not Ready Test (optional)

   Q specifies the interrupt lines.

   X - Data Interrupt Line

   Y - End-of-Operation Interrupt Line

   Z - Alarm Interrupt Line

   Note the the hexadecimal digits X, Y, and Z must correspond to the physical connections of the interrupt lines.

C. 3. Third stop, A = 0028, Q = 0000

The number in A specifies the column (3-4E in hex) to be punched in the single column pattern. The number in Q specifies the delay (in milliseconds) between cards.

4. Most loops of the program are executed 100 times. To exit a loop, set bit 15 of the stop/jump word.

D. MESSAGES

No typeouts occur if bit 8 of the stop/jump word is set.

1. Test title and initial address and frequency count typeout.

```
CRPAOD, 1728 CARD READER/PUNCH TEST
CP2F, VER. 3.1
IA = XXXX, FC = XX
```

XXXX is the initial address of the test, XX is frequency count of test.

2. Start of Section 12

SET PROTECT SWITCHES ON 1704 CONSOLE AND 1728

3. Start of Section 13

CAUSE ALARM ON 1728

4. End of 1728 test.

A	Q	A	Q
OD24	Stop/Jump word	Pass number	Return address

E. ERROR MESSAGES

1. All error messages are in the SMM17 format, e.g.,

A	Q	A	Q
ODX8	Stop/Jump word	0YZZ	Return address

where

X = number of stops (if any) or number of pairs of words typed (if any),
Y = section number,
ZZ = error code.

The section number and return address tell where in the test the error occurred. The error code indicates the type of error. Additional information will be displayed, depending on the type of error, if X is greater than 2.

2. Types of errors

Error Code	Meaning
01	Equipment address in error (operator error) Test must be called again.
02	Insufficient memory for test.
03	Parameter in error (operator error) Parameters must be selected again.
04	Unexpected internal reject.
05	Unexpected external reject.
06	Unexpected reply.
07	Unexpected Level 1 status. Additional information: A Actual status Q Expected status
08	Unexpected Level 2 status. Additional information: A Actual status Q Expected status
09	No data interrupt when expected. Additional information: A Level 1 status Q 0000
0A	Unexpected data interrupt. Additional information: A Level 1 status upon interrupt Q 0000
0B	No End-of-Operation interrupt when expected. Additional information: A Level 1 status Q 0000
0C	Unexpected End-of-Operation interrupt. Additional information: A Level 1 status upon interrupt Q 0000

I. E. 2. 0D No alarm interrupt when expected.

Additional information:

A	Q
Level 1 status	Level 2 status

0E Unexpected alarm interrupt.

Additional information:

A	Q
Level 1 status upon interrupt	Level 2 status upon interrupt

0F Unexpected 1706 Buffered Data Channel status

Additional information:

A	Q
Actual status	Expected status

10 Data error in card just read.

Additional information:

A	Q
Actual data	Expected data
Column number	Pattern number

Where the patterns are

Pattern 0 - Two-five (555, AAA, 555, AAA, etc.)

Pattern 1 - Shifted one (001, 002, 004, ---, 800, 001, 002, etc.)

Pattern 2 - Shifted zero (FFF, FFD, FFB, ---, 7FF, FFE, FFD, etc.)

Pattern 3 - "Random". Column 1 = Column 2 = ADDEND.
Word (N+1) = Word (N) + ADDEND.

Pattern 4 - Single-column. All columns blank except selected column, 79, 80. Selected column, Column 79 = 801. Column 80 = 004.

Pattern 5 - End-of-File card. Column 1 = 003. Rest of card blank (no parity or hole count bits).

Pattern 6 - Unidentified pattern.

Pattern 0 through 4 have even parity in column 79 and hole count in column 80.

Note: A blank card (all 0's) qualifies as a random card.

Note: If a data error occurs in one of columns 1 thru 78, no more of columns 1 through 78 will be checked unless Stop/Jump bit 4 is set.

I. E. 2. 11 Wrong column count

A	Q
Actual Count	Expected count (80 decimal)

12 Unidentifiable pattern.

Additional information:

A	Q
First column	Second column

Column 79 (parity) and column 80 (hole count) will be checked.

13 Wrong 1706 current word address.

Additional information:

A	Q
Actual address	Expected address

14 Unidentifiable interrupt

A	Q
Level 1 status	Not applicable

II. TEST DESCRIPTION

00. Initialization

a. (INITIAL). Type title and initial address.

b. (INITD). Determine whether legal equipment address. Error code 1 if not.

c. (INITA). Determine whether sufficient memory. Error code 2 if not.

d. Parameter stop. Error code 3 if parameter error.

0. Section 0 - PUNCH STATIC CHECK

a. Initialization

1) (SEC0). Go to Section 1 if parameter bit is not set.

2) Initialize section.

b. Section 0, Loop 0.

1) (LOOP00). Input station 0, Level 1 status.

2) Expect reply (hang on reject). Error code 4 (internal reject) or error code 5 (external reject) if not.

3) (S00A). Expect ready status. Error code 7 (unexpected level 1 status) if not.

II. 0. b. 4) Exit loop if Stop/Jump 15 is set. Loop 100 time if not. Loop indefinitely if Stop/Jump 4 is set.

c. Section 0, Loop 1

1) (LOOP01). Input station 3, Level 1 status.

2) Expect reply (hang on reject). Error code 4 or 5 if not.

3) (S01A). Expect Ready status. Error code 7 if not.

4) Loop 100 times.

d. Section 0, Loop 2

1) (LOOP02). Select all functions on station 0, expect internal reject.

2) Error code 6 if reply.

3) Error code 5 if external reject and not on 1706.

4) (S02A). If on 1706 expect 1706 status to be not reply, not reject, and Busy. Error code F if not. Terminate buffer.

5) (S02B). Get station 0, Level 1 status. Expect ready. Error 7 if not.

6) Loop 100 times.

e. Section 0, Loop 3.

1) (LOOP03). Select all functions on station 2, expect internal reject.

2) Error code 6 if reply.

3) Error code 5 if external reject and not on 1706.

4) (S03A). If on 1706 expect 1706 status to be not reply, not reject, and Busy. Error code F if not. Terminate buffer.

5) (S03B). Get station 0 status. Expect Ready. Error 7 if not.

6) Loop 100 times.

f. Section 0, Loop 4.

1) (LOOP04). Input level 1, station 2 status. Expect reply (hang on reject). Error code 4 or 5 if not.

2) (S04A). Expect Ready and Data status. Error 7 if not.

3) Loop 100 times.

g. Section 0, Loop 5.

1) (LOOP05). Input level 2, Station 2 status. Expect reply (hang on reject). Error 4 or 5 if not.

2) (S05A). Expect Zero status. Error 8 if not.

3) Loop 100 times.

h. Section 0, Loop 6.

1) (LOOP06). Clear controller on station 2. 'Expect reply (hang on reject). Error 4 or 5 if not.

2) (S06A). Get station 2, Level 1 status. Expect Data and Ready status. Error 7 if not.

3) Loop 100 times.

i. Section 0, Loop 7.

1) (LOOP07). Clear interrupts on station 2. Expect reply (hang on reject). Error 4 or 5 of not.

2) (S07A). Get station 2 level 1 status. Expect Data and Ready status. Error 7 if not.

3) Loop 100 times.

j. Section 0, Loop 8.

1) (LOOP08). Set interrupt mask bits for all three interrupts.

2) (S08A). Request data interrupt on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

3) (S08B). Expect data interrupt. Error 9 if none.

4) Expect no End-of-Operation interrupt. Error C if EOP interrupt.

5) Expect no alarm interrupt. Error E if alarm interrupt.

6) Get station 2, Level 1 status. Expect Data, Interrupt, and Ready status. Error 7 if not.

II. 0. j. 7) (S08C). Clear controller on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

8) (S08D). Loop 100 times.

k. Section 0, Loop 9.

1) (LOOP09). Set mask bits for all three interrupts.

2) (S09A). Request EOP interrupt on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

3) (S09B). Error A if data interrupt occurs.

4) Error C if EOP interrupt occurs.

5) Error E if alarm interrupt occurs.

6) Get station 2, Level 1 status. Expect Data and Ready status. Error 7 if not.

7) (S09C). Clear controller on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

8) (S09D). Loop 100 times.

l. Section 0, Loop 10.

1) (LOOP0A). Set mask bits for all three interrupts.

2) (S0BA). Get station 2, Level 1 status. Expect Data and Ready. Error 7 if not.

3) Loop 100 times.

n. Section 0, Loop 12.

1) (LOOP0C). Select all undefined functions on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

2) (S0CA). Get station 2, Level 1 status. Expect Data and Ready. Error 7 if not.

3) Loop 100 times.

o. Section 0, Loop 13.

1) (LOOP0D). Set mask bits for all interrupts.

2) (S0DA). Select all functions except Feed on station 2. Expect reply (hand on reject). Error 4 or 5 if not.

3) (S0DB). Expect data interrupt. Error 9 if none.

4) Error C if EOP interrupt occurs.

5) Error E if alarm interrupt occurs.

6) Get station 2, Level 1 status. Expect Data, Interrupt, and Ready status. Error 7 if not.

7) (S0DC). Clear controller on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

8) (S0DD). Loop 100 times.

p. End of Section 0.

1) (SEC0A). End-of-Section 0.

2) (SEC0B). Repeat section if Stop/Jump 5 is set.

1. <u>Section 1 - READER STATIC CHECK</u>

a. Initialization

1) (SEC1). Go to Section 2 if parameter bit is not set.

2) Initialize Section 1.

b. Execute Section 0, Loops 4 through 13 using station 1 instead of station 2 for all functions and status inputs. Expect status to be Ready instead of Data and Ready. Expect No Data interrupt and No Interrupt status upon data interrupt requests in Loops 8 and 13.

c. Execute Section 1, Loop 14.

1) (LOOP0E). Attempt data input on station 1. Expect external reject. Error 6 or 4 if not.

2) Get Station 1 status. Expect Ready status. Error 7 if not.

3) Loop 100 times.

d. End of Section 1.

1) (SEC0A). End-of-Section stop if Stop/Jump 1 is set.

2) (SEC0B). Repeat section if Stop/Jump 5 is set.

II. 2. Section 2 - PUNCH FEED, INTERRUPT CHECK

a. Initialization

1) (SEC2). Go to Section 3 if parameter bit is not set.

2) Initialize section.

b. Section 2, Loop 0.

1) (LOOP20). Initialize column counter.

2) Delay

3) (MOTOR). Increment loop counter.

4) (MOTORA). Select feed and clear on reader. Expect reply (hang on reject). Error 4 or 5 if not.

5) (MOTORB). Get station 1 status. Expect Busy and Ready. Error 7 if not.

6) (MOTORC). Wait for reader Not Busy.

7) (MOTORD). Clear controller on station 1. Expect reply (hang on reject). Error 4 or 5 if not.

8) (S20A). Select feed, clear, and all interrupts on station 2. Expect reply (hang on reject). Error 4 or 5 if not.

9) (S20D). Wait for column 1 data status.

10) (S20G). Expect status to be Data, Interrupt, and Ready. Error 7 if not.

11) Expect data interrupt. Error 9 if none.

12) (S20GA). Expect no alarm interrupt. Error E if alarm interrupt.

13) Restart loop if alarm interrupt and Stop/Jump 15 is clear. Exit loop if alarm interrupt and Stop/Jump 15 set.

14) (S20H). Output data (zeros) on station 2. Expect reply (restart loop if not). Error 4 or 5 if not.

15) (S20C). Get station 2 status. Expect Busy and Ready. Error 7 if not.

16) Execute steps 9) through 15) for columns 2 through 80. Wait 9 milliseconds maximum for data status in step 9). Expect status to be Data, Interrupt, Busy, and Ready in step 9).

17) (S20D). Wait 9 milliseconds maximum for EOP status.

18) (S20E). Expect status to be EOP, Interrupt, and Ready. Error 7 is not.

19) (S20I). Error B if not EOP interrupt.

20) (S20J). Clear interrupts on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

21) (S20K). Get Station 2 status. Expect EOP and Ready status. Error 7 if not.

22) (S20L). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

23) (S20M). Get Station 2 status. Expect EOP and Ready status. Error 7 if not.

24) (S20N). Loop 50 times.

25) Offset last card.

c. End of Section 2.

1) (SEC2A). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (SEC2B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

3. <u>Section 3 - READER FEED, INTERRUPT CHECK</u>

a. Initialization

1) (SEC3). Go to Section 4 if parameter bit is not set.

2) Initialize section.

b. Section 3, Loop 0.

1) (LOOP30). Delay

2) Feed a card on the reader to start the motor. Wait Not Busy.

II. 3. b. 3) (S30A). Select Feed, Clear, and all interrupts on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

4) (S30B). Set mask bits for all three interrupts.

5) (S30C). Get station 1 status. Expect Busy and Ready.. Error 7 if not.

6) (S30D). Wait 32 milliseconds maximum for column 1 data status.

7) (S30F). Expect status to be Data, Interrupt, Busy, and Ready. Error 7 if not.

8) Error E if alarm interrupt.

9) Input data on station 1. Expect reply. Error 4 or 5 if not.

10) Execute steps 5) through 9) for 80 columns. In step 6) wait about 700 microseconds maximum for data status.

11) (S30D). Wait about 700 microseconds maximum for EOP status.

12) (S30DA). Expect EOP interrupt, Ready status. Error 7 if not.

13) (S30G). Error B if not EOP interrupt.

14) (S30H). Clear interrupt on station 1. Expect Reply (hang on reject). Error 4 or 4 if not.

15) (S30I). Get station 1 status. Expect EOP and Ready status. Error 7 if not.

16) (S30J). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

17) (S30K). Get station 1 status. Expect EOP and Ready status. Error 7 if not.

18) Loop 50 times.

c. Section 3, Loop 1 - Force Lost Data

1) (LOOP31). Delay

2) Feed a card and wait Not Busy to start motor.

3) (S31A). Select Feed, Clear, EOP and Alarm interrupts. Expect Reply (hang on reject). Error 4 or 5 if not.

4) (S31B). Get station 1 status. Expect Busy and Ready. Error 7 if not.

5) (S31C). Wait 120 milliseconds maximum for EOP interrupt.

6) Error B if no EOP interrupt.

7) (S31D). Expect status upon EOP interrupt to be lost data alarm, EOP, Data, Interrupt, Not Busy, Ready. Error 7 if not.

8) (S31E). Error D if no alarm interrupt.

9) (ALA13). When alarm interrupt occurs, get status and save. Clear interrupts and select EOP interrupt.

10) (S31F). Expect status upon alarm interrupt to be Lost Data, Alarm, Data, Interrupt, Busy, Ready. Error 7 if not.

11) (S31G). Clear interrupts on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

12) (S31H). Get station 1 status. Expect lost data, Alarm, EOP, Data, Ready. Error 7 if not.

13) (S31I). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

14) (S31J). Get station 1 status. Expect EOP and Ready. Error 7 if not.

15) Loop 50 times.

d. End of Section 3.

1) (SEC3A). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (SEC3B). End-of-Section stop.

3) (SEC3C). Repeat section if Stop/Jump 5 is set.

4. Section 4 - PUNCH WHEN NO REJECT

a. Initialization

1) (SEC4). Go to section 5 if parameter bit is not set.

2) Initialize section.

3) Generate pattern 0 (two-five) in output buffer area.

II. 4. b. Section 4, Loop 0.

1) (LOOP40). Initialize column count.

2) Delay.

3) (S40A). Select Feed, Clear, EOP and Alarm interrupts. Expect Reply (hang on reject). Error 4 or 5 if not.

4) (S40D). Attempt data output on station 2. Error 4 if internal reject. Try again if external reject.

5) (S40E). Get station 3 status. Expect Busy and Ready status. Error 7 if not.

6) Error E if alarm interrupt. Offset bad card.

7) Error C if EOP interrupt.

8) Execute steps 4) through 7) for 80 columns.

9) (S40F). Wait for EOP interrupt.

10) Get station 3 status. Expect EOP, Interrupt, Ready. Error 7 if not.

11) Loop 100 times.

12) (S40FA). Generate pattern 5 (End-of-File).

13) Punch End-of-File card.

14) (S40G). Offset End-of-File card.

c. End of Section 4.

1) (SEC4A). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (SEC4B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

5. <u>Section 5 - PUNCH WHEN DATA STATUS</u>

a. Initialization

1) (SEC5). Go to Section 6 if parameter bit is not set.

2) Initialize section.

3) Generate pattern 1 (shifting 1) in output buffer area.

b. Section 5, Loop 0.

1) (LOOP50). Initialize column count.

2) Delay.

3) (S50A). Select Feed, Clear, Alarm and EOP interrupts on station 2.

4) (S50C). Wait for data status.

5) Expect status to be Data and Ready. Error 7 if not.

6) (S50D). Output data on station 2.

7) (S50E). Get station 2 status. Expect Busy and Ready status. Error 7 if not.

8) (S50EA). Error E if alarm interrupt. Offset bad card.

9) Error C if EOP interrupt.

10) Execute steps 4) thru 9) for 80 columns. In step 5) except status to be Data, Busy, and Ready for columns 2 through 80.

11) (S50F). Wait for EOP interrupt.

12) Get station 2 status. Expect EOP, Interrupt, and Ready. Error 7 if not.

13) Loop 100 times.

14) (S50FA). Generate pattern 5 (End-of-File).

15) Punch End-of-File card.

16) (S50G). Offset End-of-File card.

c. End of Section 5.

1) SEC5A). Clear controller on station 2.

2) (SEC5B). End-of-Section stop.

3) Repeat section if Stop/Jump bit 5 is set.

6. Section 6 - PUNCH WHEN DATA INTERRUPT

a. Initialization

1) (SEC6). Go to Section 7 if parameter bit is not set.

2) Initialize section.

II. 6. a. 3) Generate pattern 2 (shifting 0) in output buffer area.

b. Section 5, Loop 0.

1) (LOOP60). Initialize column counter.

2) Delay.

3) (S60A). Select Feed, Clear, and all interrupts on station 2.

4) (DATA16). When data interrupt occurs, output data. Increment column counter. Do not clear interrupts.

5) (S60C). Wait for EOP or alarm interrupt.

6) Error E if alarm interrupt.

7) Error 11 if column count is not equal to 80.

8) (S60D). Get station 2 status. Expect EOP, Interrupt, Ready. Error 7 if not.

9) (S60DC). Loop 100 times.

10) (S60DB). Generate pattern 5 (End-of-File).

11) Punch End-of-File card.

12) (S60E). Offset End-of-File card.

c. End of Section 6.

1) (SEC6B). Clear controller on station 2.

2) (SEC6C). End-of-Section stop.

3) (SEC6D). Repeat section if Stop/Jump 5 is set.

7. Section 7 - BUFFERED OUTPUT

a. Initialization

1) (SEC7). Go to Section 8 if parameter bit is not set.

2) Initialize section

3) (SEC7A). If 1728 is not on 1706 (W=0), execute Section 6 with a new random pattern for each pass through Section 6, Loop 0, unless Stop/Jump 12 is set. In that case use the same random pattern for each pass. Also, punch single-column pattern in Section 6.

b. Section 7, Loop 0.

1) (LOOP70). Store first word address +80 in first word address -1 of output buffer area.

2) (S70A). Select Feed, Clear, EOP and Alarm interrupts on station 2.

3) (S70B). Initiate buffered output on 1706.

4) (S70C). Wait 9 milliseconds maximum after 1706 becomes Not Busy for EOP interrupt.

5) Error B if no EOP interrupt.

6) (S70D). Expect 1706 current word address upon EOP interrupt to be first word address +80. Error 13 if not.

7) (S70E). Get station 2 status. Expect EOP, Interrupt, and Ready. Error 7 if not.

8) Error E if Alarm interrupt.

9) (S70EA). Loop 100 times. Generate new random pattern each time unless Stop/Jump 12 is set.

10) (S70EB). Generate pattern 5 (end-of-file).

11) Punch end-of-file card.

12) (S70F). Offset end-of-file card.

c. Initialization of Section 7, Loop 1.

1) Store first word address +81 in first word address -1 of output buffer area.

2) Generate pattern 4 (single-column). This pattern consists of all zeros except for the selected column and columns 79 and 80.

d. Section 7, Loop 1.

1) (LOOP71). Select feed, clear, EOP and alarm interrupts on station 2.

2) (S21A). Initiate 81-word buffered output on 1706.

II. 7. d. 3) (EOP17). When EOP interrupt occurs, get 1706 status and save. Terminate buffer if Busy. Save current word address. Get 1728 status and save.

4) Error B if 1706 becomes Not Busy before EOP interrupt occurs.

5) (S71B). Expect 1706 status upon EOP interrupt to be Reject and Busy. Error F if not.

6) Expect current word address upon EOP interrupt to be first word address +80. Error 13 if not.

7) (S71C). Get station 2 status. Expect EOP, Interrupt, Ready. Error 7 if not.

8) Error E if alarm interrupt.

9) (S71CA). Loop 10 times.

10) (S71CB). Generate pattern 5 (End-of-File).

11) Punch End-of-File card.

12) (S71D). Offset End-of-File card.

e. End of Section 7.

1) (SEC7B). Clear Controller on station 2.

2) (SEC7C). End-of-Section stop.

3) Repeat Section if Stop/Jump 5 is set.

8. <u>Section 8 - READ WHEN NO REJECT</u>

a. Initialization

1) (SEC8). Go to Section 8 if parameter bit not set.

2) Initialize section.

b. Section 8, Loop 8.

1) (LOOP80). Initialize column counter.

2) Delay.

3) (S80A). Select Feed, Clear, EOP and Alarm interrupts.

4) (S80C). Error E if alarm interrupt.

5) Error C if EOP interrupt.

6) (S80D). Attempt data input. Error 4 if internal reject. Try again if external reject.

7) (S80E). Store data. Get station 1 status. Expect Busy and Ready. Error 7 if not.

8) Execute Steps 4) through 7) for 80 columns.

9) (S80G). Check data. Determine type of pattern from first two columns to generate rest of pattern. Compare actual with expected data. Offset card if data error.

10) (S80H). Wait for EOP interrupt.

11) (S80I). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

12) Go to step 1) until End-of-File card is read.

c. End of Section 8.

1) (SEC8A). Clear controller.

2) (SEC8B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

9. Section 9 - READ WHEN DATA STATUS

a. Initialization

1) (SEC9). Go to Section 10 if parameter bit is not set.

2) Initialize section.

b. Section 9, Loop 0.

1) (LOOP90). Initialize column counter.

2) Delay.

3) (S90A). Select Feed, Clear, EOP and Alarm interrupts.

4) (S90C). Error E if alarm interrupt.

5) Error C if EOP interrupt.

6) (S90D). Wait for data status.

7) Expect status to be Data, Busy, Ready. Error 7 if not.

8) (S90E). . Input data.

II. 9. b. 9) (S90F). Save data. Get station 1 status. Expect Busy and Ready. Error 7 if not.

10) Execute steps 4) through 9) for 80 columns.

11) (S90H). Check data. Offset card if data error.

12) (S90I). Wait for EOP interrupt.

13) (S90J). Expect status upon EOP interrupt to the EOP, Interrupt, and Ready, Error 7 if not.

14) Go to step 1) until End-of-File card is read.

c. End of Section 9.

1) (SEC9A). Clear controller.

2) (SEC9B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

10. Section 10 - READ WHEN DATA INTERRUPT

a. Initialization

1) (SECA). Go to Section 11 if parameter bit is not set.

2) Initialize section.

b. Section 10, Loop 0.

1) (LOOPA0). Initialize column counter.

2) Delay.

3) (SA0A). Select Feed, Clear and all interrupts.

4) (SA0C). Error E if alarm interrupt.

5) (SA0D). Wait for EOP interrupt.

6) (DATAIA). Input and store data in interrupt routine. Increment column count. Do not clear interrupts.

7) Except column count to be 80. Error 11 if not.

8) (SA0E). Check data. Offset card if data error.

9) (SA0F). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

10) Loop to 1) until End-of-File card is read.

c. End of Section 10.

1) (SECAA). Clear controller.

2) (SECAB). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

11. Section 11 - BUFFERED READ

a. Initialization

1) (SECB). Go to Section 12 if parameter bit is not set.

2) Go to Section 12 if no 1706 (W=0).

3) Initialize section.

b. Section 11, Loop 0.

1) (LOOPB0). Select Feed, Clear, EOP and Alarm interrupts.

2) (SB0A). Initiate 80-word buffered input.

3) Wait 1 millisecond after 1706 becomes Not Busy for EOP interrupt. Error B if no EOP interrupt.

4) Error E if alarm interrupt.

5) (SB0B). Expect 1706 current word address to be first word address +80. Error 13 if not.

6) (SB0C). Check data. Offset card if data error.

7) (SB0D). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

8) Loop to 1) until End-of-File card is read.

c. Initialize Section 11, Loop 1.

Store first word address +81 in first word address - 1 of input buffer area.

d. Section 11, Loop 1.

1) (LOOPB1). Select Feed, Clear, EOP and Alarm interrupts.

2) (SB1A). Initiate 81-word buffered input.

II. 11. d. 3) (EOP17). When EOP interrupt occurs, get 1706 status and save. Terminate buffer if Busy. Save current word address. Get 1728 status and save.

4) Error B if 1706 becomes Not Busy before EOP interrupt.

5) Error E if alarm interrupt.

6) (SB1B). Expect 1706 status upon EOP interrupt to be Reject and Busy. Error F if not.

7) Expect current word address upon EOP interrupt to be first word address +80. Error 13 if not.

8) (SB1C). Check data. Offset card if data error.

9) (SB1D). Expect 1728 status upon EOP interrupt to the EOP, Interrupt, Ready. Error 7 if not.

10) Loop to 1) until End-of-File card is read.

e. End of Section 11.

1) (SECBA). Clear controller.

2) (SECBB). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

12. Section 12 - PROTECTION TEST

a. Initialization

1) (SECC). Go to Section 13 if parameter bit is not set.

2) (SECCA). Clear protect bits in all memory locations through End-of-Test.

3) (SECCB). Typeout message to operator to set Protect switches on 1728 and 1704.

4) Stop with A = Q = 1728.

5) Operator must set Protect switches and hit RUN.

b. Section 12, Loop 0.

1) (LOOPC0). Get station 0 status. Expect Protected and Ready. Error 7 if not.

2) Loop 100 times.

c. Section 12, Loop 1.

1) (LOOPC1). Attempt all functions on station 1. Expect external reject. Error 4 or 6 if not.

2) Get station 1 status. Expect Protected and Ready. Error 7 if not.

3) Loop 100 times.

d. Section 12, Loop 2.

1) (LOOPC2). Attempt all functions on station 2. Expect external reject. Error 4 or 6 if not.

2) Get station 2 status. Expect Protected, Data, Ready. Error 7 if not.

3) Loop 100 times.

e. End of Section 12.

1) Set bit 1 of stop/jump word.

2) End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

13. Section 13 - ALARM WITH NOT READY TEST

a. Initialization

1) (SECD). Go to ENDTEST if parameter bit is not set.

2) (SECDA). Select Clear Controller and Alarm interrupt.

3) (SECDB). Type out message to operator to cause alarm on 1728.

4) Stop with A = Q = 1728.

5) Operator must cause alarm on 1728 and hit RUN.

b. Section 13, Loop 0.

1) (LOOPD0). Get station 0 status. Expect Alarm, Interrupt, Not Ready status. Error 7 if not.

2) Set Alarm interrupt mask bit.

II. 13. b. 3) Error D if no alarm interrupt.

4) Loop 100 times.

c. Section 13, Loop 1.

1) (LOOPD1). Select all functions on punch. Expect external reject. Error 4 or 6 if not.

2) Get station 2 status. Expect Alarm, Interrupt, Not Ready. Error 7 if not.

3) Loop 100 times.

d. Section 13, Loop 2.

1) (LOOPD2). Select all functions on reader. Expect external reject. Error 4 or 6 if not.

2) Get station 2 status. Expect Alarm, Interrupt, Not Ready. Error 7 if not.

3) Loop 100 times.

e. End of Section 13.

1) Set Stop/Jump bit 1.

2) End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

14. End of 1728 test.

(ENDTEST). Typeout End-of-Test message. Stop if Stop/Jump 2 is set. Repeat test if Stop/Jump 6 is set.

(RESTART). Re-enter parameters if Stop/Jump 10 is set.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - Approximately 3070_{10}.

B. TIMING - Variable depending upon the size of test deck.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 8K memory.
2. 1728 Card Reader
3. A device for loading test.

1729-2/3/411 CARD READER TEST

(CR3A13 Test No. 13)
(CP=OF)

I. OPERATING INSTRUCTIONS

A. RESTRICTIONS

1. Section 12, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to set the Protect switches on the 17X4 console and on the 1729-2/3. He must clear the Protect switch on the console at the end of the section for further testing.

2. Section 13, an optional section, requires manual intervention. A stop occurs near the start of the section for the operator to cause an alarm condition on the 1729-2/3 for example, by making the input hopper empty. He must clear the alarm condition at the end of the section for further testing.

3. Section 11 will not be executed unless the 1729-2/3 is on a 1706 (or 1716) Buffered Data Channel.

4. Special test decks are used for each section 8 through 11. In each of sections 8 through 11, cards are read through the end-of-file card. Cards having data errors are offset, except on 1729-3.

5. The test cannot be loaded into a 4K computer.

6. This test must be run alone.

7. SMM parameter bits 2 and 3 must be correctly set before executing this test to ensure correct timing constants.

B. LOADING PROCEDURE

Call as external test number 13 under SMM17. The equipment address must have bit 0 set and bits 1 through 6 all clear.

Restart test after loading by MC, set P = IA and run.

C. PARAMETERS

If bit 0 of the SMM Stop/Jump word is set at the start of the test or at the start of succeeding passes through the test, a parameter stop occurs. A typeout of selected parameters will occur.

1. First stop, A = 1341, Q = Stop/Jump word.

2. Second stop, A = 003F, Q = 0XYZ. The bits in the A register specify the sections to be tested, i.e., sections 0, 3, 8, 9, 10, and 11. The sections available are:

Section		A Register
0	- Reader Static Check	Bit 0
3	- Reader Feed, Interrupt Check	Bit 1

Section		A Register
8	- Read when No Reject	Bit 2
9	- Read when Data Status	Bit 3
10	- Read when Data Interrupt	Bit 4
11	- Buffered Read	Bit 5
12 (optional)	- Protection Test	Bit 6
13 (optional)	- Alarm with Not Ready Test	Bit 7

Q specifies the interrupt lines.

X - Data Interrupt Line

Y - End-of-Operation Interrupt Line

Z - Alarm Interrupt Line

Note: The hexadecimal digits X, Y and Z must correspond to the physical connections of the interrupt lines.

3. Third stop, A= 0028, Q - 0000
   The number in A specifies the column (3-4E in hex) to be expected in the single column pattern. The number in Q specifies the maximum delay (in milliseconds) between cards. This value will be decremented down to 0 then turn back to original amount in 50_{10} even increments.

4. Fourth stop, Card Reader Type (A = 0 1729-2, A = 1 1729-3)
   (Q = zero)

5. Most loops of the program are executed 100 times. To exit a loop, set bit 15 of the Stop/Jump word.

D. MESSAGES

No typeouts occur if bit 8 of the Stop/Jump word is set

1. Test title, initial address and frequency count typeout.

```
CR3A13, 1729-2/3 CARD READER TEST
  CPOF, Ver. 3.1
  IA = XXXX, FC = XX
```

XXXX is the initial address of the test, XX is the frequency count

2. Start of Section 12

SET PROTECT SWITCHES ON 17X4 CONSOLE AND 1729-2/3, HIT RUN - IF NON-STD STOP WITH A-REG EQUAL M-REG-CLEAR PROTECT SWITCHES AND HIT RUN, CHECK TYPEOUT FOR END SECT OR ERROR

3. Start of Section 13

CAUSE ALARM ON 1729-2/3, HIT RUN

4. End of 1729-2/3 test

A	Q	A	Q
1324	Stop/Jump word	Pass number	Return address

E. ERROR MESSAGES

1. All error messages are in the SMM17 format, i.e.,

A	Q	A	Q
13X8	Stop/Jump word	0YZZ	Return Address

where

X = number of stops (if any) or number of pairs of words typed (if any),
Y = section number,
ZZ = error code

The section number and return address tell where in the test the error occurred. The error code indicates the type of error. Additional information will be displayed, depending on the type of error, if X is greater than 2.

2. Types of errors.

Error Code	Meaning
01	Equipment address in error (operator error) Test must be called again
02	Insufficient memory for test
03	Parameter in error (operator error) Parameters must be selected again
04	Unexpected internal reject.
05	Unexpected external reject
06	Unexpected reply
07	Unexpected level 1 status. Additional information: A — Q Actual status — Expected status.
08	Unexpected level 2 status Additional information: A — Q Actual status — Expected status
09	No data interrupt when expected Additional information: A — Q Level 1 status — 0000

0A Unexpected data interrupt
Additional information:

A	Q
Level 1 status upon interrupt	0000

0B No End-of-Operation interrupt when expected
Additional information:

A	Q
Level 1 status	0000

0C Unexpected End-of-Operation interrupt
Additional information:

A	Q
Level 1 status upon interrupt	0000

0D No Alarm interrupt when expected
Additional information:

A	Q
Level 1 status	Level 2 status

0E Unexpected alarm interrupt
Additional information:

A	Q
Level 1 status upon interrupt	Level 2 status upon interrupt

0F Unexpected 1706 Buffered Data Channel status
Additional information:

A	Q
Actual status	Expected status

10 Data error in card just read
Additional information:

A	Q
Actual data	Expected data
Column number	Pattern number

Where the pattern is

Pattern 0 - Two-five (555, AAA, 555, AAA, etc.)

Pattern 1 - Shifted one (001, 002, 004, ---, 800, 001, 002, etc.)

Pattern 2 - Shifted zero (FFE, FFD, FFB, ---, 7FF, FFE, FFD, etc.)

Pattern 3 - "Random". Column 1 = Column 2 = ADDEND. Word (N+1) = Word (N) + ADDEND.

Pattern 4 - Single-column. All columns blank except selected column, 79, 80. Selected column, Column 79 = 801. Column 80 = 004.

Pattern 5 - End-of-File card. Column 1 = 003. Rest of card blank (no parity or hole count bits).

Pattern 6 - Unidentified pattern.

Pattern 0 through 4 have Even parity in column 79 and hole count in column 80.

Note: A blank card (all zeros) qualifies as a random card.

Note: If a data error occurs in one of columns 1 through 78, no more of columns 1 through 78 will be checked unless Stop/Jump bit 4 is set.

11 Wrong Column Count

A = Actual Count Q = Expected Count (80 decimal)

12 Unidentifiable pattern.
Additional information:

A	Q
First column	Second column

Column 79 (parity) and column 80 (hole count) will be checked.

13 Wrong 1706 current word address
Additional information:

A	Q
Actual address	Expected address

14 Unidentifiable interrupt

A	Q
Level 1 status	0000

II. TEST DESCRIPTION

00. Initialization

a. (INITIAL). Type title and initial address.

b. (INITD). Determine whether legal equipment address. Error code 1 if not.

c. (INITA). Determine whether sufficient memory. Error code 2 if not.

d. Parameter stop. Error code 3 if parameter error.

0. Section 0 - READER STATIC CHECK

a. Initialization

1) (SECO). Go to Section 3 if parameter bit is not set.

2) Initialize section.

b. Section 0, Loop 0.

1) (L00P00). Input station 0, level 1 status.

2) Expect Reply (hang on reject). Error code 4 (internal reject) or error code 5 (external reject) if not.

3) (S00A). Expect Ready status. Error code 7 (unexpected level 1 status) if not.

4) Exit loop if Stop/Jump 15 is set. Loop 100 times if not. Loop indefinitely if Stop/Jump 4 is set.

c. Section 0, Loop 1

1) (L00P01). Input station 3, level 1 status.

2) Expect Reply (hang on reject). Error code 4 or 5 if not.

3) (S01A). Expect Ready status. Error code 7 if not.

4) Loop 100 times.

d. Section 0, Loop 2

1) (L00P02). Select all functions on station 0, expect internal reject.

2) Error code 6 if reply.

3) Error code 5 if external reject and not on 1706.

4) (S02A). If on 1706 expect 1706 status to be Not Reply, Not Reject, and Busy. Error code F if not. Terminate buffer.

5) (S02B). Get station 0, level 1 status. Expect Ready. Error 7 if not.

6) Loop 100 times.

e. Section 0, Loop 3.

   1) (L00P03). Select all functions on station 3, expect internal reject.

   2) Error code 6 if reply.

   3) Error code 5 if external reject and not on 1706.

   4) (S03A). If on 1706 expect 1706 status to be Not Reply, Not Reject, and Busy. Error code F if not. Terminate buffer.

   5) (S03B). Get station 0 status. Expect Ready. Error 7 if not.

   6) Loop 100 times.

f. Section 0, Loop 4.

   1) (L00P04) Input level 1, station 1 status. Expect Reply (hang on reject). Error code 4 or 5 if not.

   2) (S04A). Expect Ready and Data status. Error 7 if not.

   3) Loop 100 times.

g. Section 0, Loop 5.

   1) (L00P05). Input level 2, station 1 status. Expect Reply (hang on reject). Error 4 or 5 if not.

   2) (S05A). Expect zero status. Error 8 if not.

   3) Loop 100 times.

h. Section 0, Loop 6.

   1) (L00P06). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

   2) (S06A). Get station 1, level 1 status. Expect Data and Ready status. Error 7 if not.

   3) Loop 100 times.

i. Section 0, Loop 7.

   1) (L00P07). Clear interrupts on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

   2) (S07A). Get station 1 level 1 status. Expect Data and Ready status. Error 7 if not.

   3) Loop 100 times.

j. Section 0, Loop 8.

1) (L00P08). Set interrupt mask bits for all three interrupts.

2) (S08A). Request Data interrupt on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

3) (S08B). Expect Data interrupt. Error 9 if none.

4) Expect no End-of-Operation interrupt. Error C if EOP interrupt.

5) Expect no Alarm interrupt. Error E if Alarm interrupt.

6) Get station 1, level 1 status. Expect Data, Interrupt, and Ready status. Error 7 if not.

7) (S08C). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

8) (S08D). Loop 100 times.

k. Section 0, Loop 9

1) (L00P09). Set mask bits for all three interrupts.

2) (S09A). Request EOP interrupt on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

3) (S09B). Error A if Data interrupt occurs.

4) Error C if EOP interrupt occurs.

5) Error E if Alarm interrupt occurs.

6) Get station 1, level 1 status. Expect Data and Ready status, error 7 if not.

7) (S09C). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

8) (S09D). Loop 100 times.

l. Section 0, Loop 10.

1) (L00P0A). Set mask bits for all three interrupts.

2) (S0AA). Request Alarm interrupt on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

3) (S0AB). Error A if Data interrupt occurs..

4) Error C if EOP interrupt occurs.

5) Error E if Alarm interrupt occurs.

6) Get station 1, level 1 status. Expect Data and Ready. Error 7 if not.

7) (S0AC). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

8) (S0AD). Loop 100 times.

m. Section 0, Loop 11.

1) (L00P0B). Select offset on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (S0BA). Get station 1, level 1 status. Expect Data and Ready. Error 7 if not.

3) Loop 100 times.

n. Section 0, Loop 12.

1) (L00P0C). Select all undefined functions on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (S0CA). Get station 1, level 1 status. Expect Data and Ready. Error 7 if not.

3) Loop 100 times.

o. Section 0, Loop 13.

1) (L00P0D). Set mask bits for all interrupts.

2) (S0DA). Select all functions except Feed on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

3) (S0DB). Expect data interrupt. Error 9 if none.

4) Error C if EOP interrupt occurs.

5) Error E if alarm interrupt occurs.

6) Get station 1, level 1 status. Expect Data, Interrupt, and Ready status. Error 7 if not.

7) (S0DC). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

8) (S0DD). Loop 100 times.

p. End of Section 0.

1) (SEC0A). End-of-Section 0.

2) (SEC0B). Repeat section if Stop/Jump 5 is set.

3. Section 3 - READER FEED, INTERRUPT CHECK

a. Initialization

1) (SEC3). Go to Section 8 if parameter bit is not set.

2) Initialize section.

b. Section 3, Loop 0.

1) Delay

2) Feed a card on the reader to start the motor. Wait Not Busy.

3) (S30A). Select Feed, Clear, and all interrupts on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

4) (S30B). Set mask bits for all three interrupts.

5) (S30C). Get station 1 status. Expect Busy and Ready. Error 7 if not.

6) (S30D). Wait 40 milliseconds for column 1 data status.

7) (S30F). Expect status to be Data, Interrupt, Busy, and Ready. Error 7 if not.

8) Error E if alarm interrupt.

9) Input data on station 1. Expect reply. Error 4 or 5 if not.

10) Execute steps 4) thru 8) for 80 columns. In step 5) wait about 1200 microseconds for data status.

11) (S30D). Wait about 1200 microseconds for EOP status.

12) (S30DA). Expect EOP interrupt, Ready status. Error 7 if not.

13) (S30G). Error B if no EOP interrupt.

14) (S30H). Clear interrupt on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

15) (S301). Get station 1 status. Expect EOP and Ready status. Error 7 if not.

16) (S30J). Clear controller on station 1. Expect reply (hang on reject). Error 4 or 5 if not.

17) (S30K). Get station 1 status. Expect EOP and Ready status. Error 7 if not.

18) Loop 50 times.

c. Section 3, Loop 1 - Force Lost Data

1) Delay

2) Feed a card and wait Not Busy to start motor.

3) (S31A). Select Feed, Clear, EOP and Alarm interrupts. Expect Reply (hang on reject). Error 4 or 5 if not.

4) (S31B). Get station 1 status. Expect Busy and Ready. Error 7 if not.

5) (S31C). Wait XXX milliseconds for EOP interrupt.

6) Error B if no EOP interrupt.

7) (S31D). Expect status upon EOP interrupt to be Lost Data, Alarm, EOP, Data, Interrupt, Not Busy, Ready. Error 7 if not.

8) (S31E). Error D if no alarm interrupt.

9) (ALA13). When alarm interrupt occurs, get status and save. Clear interrupts and select EOP interrupt.

10) (S31F). Expect status upon alarm interrupt to be Lost Data, Alarm, Data, Interrupt, Busy, Ready. Error 7 if not.

11) (S31G). Clear interrupts on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

12) (S31H). Get station 1 status. Expect Lost Data, Alarm, EOP, Data, Ready. Error 7 if not.

13) (S311). Clear controller on station 1. Expect Reply (hang on reject). Error 4 or 5 if not.

14) (S31J). Get station 1 status. Expect EOP and Ready. Error 7 if not.

15) Loop 50 times.

d. End of Section 3

1) (SEC3A). Clear controller on station 2. Expect Reply (hang on reject). Error 4 or 5 if not.

2) (SEC3B). End-of-Section stop.

3) (SEC3C). Repeat section if Stop/Jump 5 is set.

8. Section 8 - READ WHEN NO REJECT

a. Initialization

1) (SEC8). Go to Section 9 if parameter bit not set.

2) Initialize section

b. Section 8, Loop 0

1) (L00P80). Initialize column counter.

2) Delay

3) (S80A). Select Feed, Clear, EOP and Alarm interrupts.

4) (S80C). Error E if Alarm interrupt.

5) Error C if EOP interrupt.

6) (S80D). Attempt data input. Error 4 if internal reject. Try again if external reject.

7) (S80E). Store data. Get station 1 status. Expect Busy and Ready. Error 7 if not.

8) Execute Steps 4) through 7) for 80 columns.

9) (S80C).. Check data. Determine type of pattern from first two columns to generate reset of pattern. Compare actual with expected data. Offset card if data error.

10) (S8011). Wait for EOP interrupt.

11) (S801). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

12) Go to step 1) until End-of-File card is read.

c. End of Section 8

1) (SEC8A). Clear controller.

2) (SEC8B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

9. Section 9 - READ WHEN DATA STATUS

a. Initialization

1) (SEC9). Go to Section 10 if parameter bit is not set.

2) Initialize section.

b. Section 9, Loop 0.

1) (L00P90). Initialize column counter.

2) Delay

3) (S90A). Select Feed, Clear, EOP and Alarm interrupts.

4) (S90C). Error E if Alarm interrupt.

5) Error C if EOP interrupt.

6) (S90D). Wait for data status.

7) Expect status to be Data, Busy, Ready. Error 7 if not.

8) (S90E). Input data.

9) (S90F). Save data. Get station 1 status. Expect Busy and Ready. Error 7 if not.

10) Execute steps 4) through 9) for 80 columns.

11) (S90H). Check data. Offset card if data error.

12) (S90I). Wait for EOP interrupt.

13) (S90J). Expect status upon EOP interrupt to be EOP, Interrupt, and Ready. Error 7 if not.

14) Go to step 1) until End-of-File card is read.

c. End of Section 9

1) (SEC9A). Clear controller.

2) (SEC9B). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

10. Section 10 - READ WHEN DATA INTERRUPT

a. Initialization

1) (SECA). Go to Section 11 if parameter bit is not set.

2) Initialize section.

b. Section 10, Loop 0

1) (L00PA0). Initialize column counter.

2) Delay

3) (SA0A). Select Feed, Clear and all interrupts.

4) (SA0C). Error E if alarm interrupt.

5) (SA0D). Wait for EOP interrupt.

6) (DATAIA). Input and store data in interrupt routine. Increment column count. Do not clear interrupts.

7) Expect column count to be 80. Error 11 if not.

8) (SA0E). Check data. Offset card if data error.

9) (SA0F). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

10) Loop to 1) until End-of-File card is read.

c. End of Section 10

1) (SECAA). Clear controller.

2) (SECAB). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

11. Section 11 - BUFFERED READ

a. Initialization

1) (SECB). Go to Section 12 if parameter bit is not set.

2) Go to Section 12 if no 1706 (W = 0).

3) Initialize section.

b. Section 11, Loop 0

1) (L00PB0). Select Feed, Clear, EOP and Alarm interrupts.

2) (SB0A). Initiate 80-word buffered input.

3) Wait 1 millisecond after 1706 becomes Not Busy for EOP interrupt. Error B if no EOP interrupt.

4) Error E if alarm interrupt.

5) (SBOB). Expect 1706 current word address to be first word address + 80. Error 13 if not.

6) (SB0C). Check data. Offset card if data error.

7) (SB0D). Expect status upon EOP interrupt to be EOP, Interrupt, Ready. Error 7 if not.

8) Loop to 1) until End-of-File card is read.

c. Initialize Section 11, Loop 1

Store first word address + 81 in first word address - 1 of input buffer area.

d. Section 11, Loop 1

1) (L0OPB1). Select Feed, Clear, EOP and Alarm interrupts.

2) (SB1A). Initiate 81-word buffered input.

3) (E0P17). When EOP interrupt occurs, get 1706 status and save. Terminate buffer if Busy. Save current word address. Get 1729-2/3 status and save.

4) Error B if 1706 becomes Not Busy before EOP interrupt.

5) Error E if alarm interrupt.

6) (SB1B). Expect 1706 status upon EOP interrupt to be Reject and Busy. Error F if not.

7) Expect current word address upon EOP interrupt to be first word address + 80. Error 13 if not.

8) (SB1C). Check data. Offset card if data error.

9) (SB1D). Expect 1729-2/3 status upon EOP interrupt to be EOP, interrupt, ready. Error 7 if not.

10) Loop to 1) until End-of-File card is read.

e. End of Section 11

1) (SECBA). Clear controller.

2) (SECBB). End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

12. Section 12 - PROTECTION TEST

a. Initialization

1) (SECC). Go to section 13 if parameter bit is not set.

2) (SECCA). Clear protect bits in all memory locations through End-of-Test.

3) (SECCB). Typeout message to operator to set Protect switches on 1729-2/3 and 17X4.

4) Stop with A = 1729
Q = .2/3

5) Operator must set Protect switches and hit RUN.

b. Section 12, Loop 0

1) (L00PC0). Get station 0 status. Expect Protected and Ready. Error 7 if not.

2) Loop 100 times.

c. Section 12, Loop 1

1) (L00PC1). Attempt all functions on station 1. Expect external reject. Error 4 or 6 if not.

2) Get station 1 status. Expect Protected and Ready. Error 7 if not.

3) Loop 100 times.

d. End of Section 12

1) Set bit 1 of Stop/Jump word.

2) End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

13. Section 13 - ALARM WITH NOT READY TEST

a. Initialization

1) (SECD). Go to ENDTEST if parameter bit is not set.

2) (SECDA). Select clear controller and alarm interrupt.

3) (SECDB). Type out message to operator to cause alarm on 1729-2/3.

4) Stop with A = Q = 1729-2/3.

5) Operator must cause alarm on 1729-2/3 and hit RUN.

b. Section 13, Loop 0

1) (L00PD0). Get station 0 status. Expect Alarm, Interrupt, Not Ready status. Error 7 if not.

2) Set alarm interrupt mask bit.

3) Error D if no alarm interrupt.

4) Loop 100 times.

c. Section 13, Loop 2

1) (L00PD2). Select all functions on reader. Expect external reject. Error 4 or 6 if not.

2) Get station 1 status. Expect Alarm, Interrupt, Not Ready. Error 7 if not.

3) Loop 100 times.

d. End of Section 13

1) Set Stop/Jump bit 1.

2) End-of-Section stop.

3) Repeat section if Stop/Jump 5 is set.

14. End of 1729-2/3 test

(ENDTEST). Typeout End-of-Test message. Stop if Stop/Jump 2 is set. Repeat test if Stop/Jump 6 is set.

(RESTART). Re-enter parameters if Stop/Jump 10 is set.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - Approximately 6K.

B. TIMING - Variable depending upon the size of test deck.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 8K memory.
2. 1729-2/3 Card Reader.
3. A device for loading the test.

1726 CARD READER TEST

(CR2017 Test No. 17)
(CPOF)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

This test requires a 1700 having 8K (or more) core. Future versions of SMM17 should be capable of executing this program in a 4K machine.

B. LOADING PROCEDURE

The test operates as a subprogram under control of the 1700 System Maintenance Monitor (SMM17). The calling sequence is that specified by SMM17. This test is number 17 (hexadecimal) on the SMM17 tape. To restart test after loading MC, set P = IA and RUN.

C. PARAMETER STOPS

At the beginning of the test, the parameter stop below will occur unless Jump 6 has been selected. The parameter stop is a four stop set:

First stop: (A) = ID word (overflow is lit)
(Q) = Stop/Jump parameter

Second stop: (A) = Section select parameter
Set bit X to run section X. All sections will be run if not otherwise specified.
(Q) = Equipment, channel type, and director code. Set (Q) exactly as it would be for a status input on the 1726. (Include W, E, S, and D - W = bits 11-15, E = bits 7-10, S and D = bits 0-6).

Third stop: (A) = Data interrupt line
(Q) = Alarm interrupt line

Fourth stop: (A) = End of Operation interrupt line
(Q) = Normally is 0011 hexadecimal. If the operator does not wish to test End of File status at end of test, set the lower 8 bits of Q to 0. If these bits of Q are left nonzero, the operator should be sure the EOF switch is set on the console.

After last stop, selected parameters will be typed out.

D. TEST DECK

The test deck used in checking the 1726/405 card reader is to be structured as below:

1. Hollerith cards (any number)
2. 1 separator card with column 1 containing a 0123456789 punch, and all other columns blank
3. 25 cards (any number)
4. 1 blank binary card - (79 punch in column 1 only)
5. 52 cards (any number)
6. 1 separator card - blank, except for column 1, which contains a 0123456789 punch
7. Separator cards (any number) - 6789 punch in column 1 only.
8. 1 blank Hollerith card
9. Random cards (any number)
10. Blank cards (any number - should not exceed 240 cards, the capacity of the secondary receiving tray).
11. 2 separator cards.

II. MESSAGES

A. NORMAL

1. Title typeout:

   Typeout message: CR2017, 1726/405 CARD READER TEST.
   IA = XXXX, FC = XX, CPOF, VER. 3.1

   There is no corresponding stop.

2. Instructions to reload cards in input tray

   Typeout message (typical) - RELOAD INPUT TRAY
   HOLLERITH CARDS

   The computer will come to a select stop, with A = 3333 hex to allow the operator to reload his cards.

3. Instruction to set/clear Protect switches

Typeout messages:

SET CONSOLE AND 1726 PROTECT SWITCHES
CLR CONSOLE AND 1726 PROTECT SWITCHES

Note: If there is no teletype present, or if the Stop/Jump parameter has been chosen to omit typeouts, the computer will come to a select stop both when we are to set the switches and when we are to clear them. (A) = 0123 hex when stop for setting of Protect switches; (A) = 3210 hex when stop for clearing them.

B. ERROR

Format: All error typeouts/halts will type out in the standard SMM17 error typeout format, namely:

A	Q	A	Q	A	Q	. . .
ID	STJP	Sect. # and err. code	Rtn. Add.	Act.	Exp.	

The corresponding error stops are:

Stop 1 - (A) = ID word
(Q) = Stop/Jump parameter

Stop 2 - (A) = XXYY, where XX = section number, YY = error code.
(Q) = Return address

Stop 3 - (A) = Actual
& 4 - (Q) = Expected

See individual typeouts for what the actual and expected represent. Individual errors (all are 2 stop messages unless otherwise noted).

Error Code	Comments
01	Unexpected data interrupt occurred (similar to 03 below).
02	Spare.
03	Unexpected End of Operation interrupt occurred. This may be caused by quite a few errors, including failure of Clear Interrupt or Clear Controller functions, occurrence of End of Operation interrupt on columns other than the last one on a card, by the interrupt occurring when not selected, or another interrupt occurring, but having its status incorrect.
04	Spare.
05	Unexpected alarm interrupt occurred (similar to 03 above).
06	Spare.
07	Clear controller failed to clear binary status.
08	No expected End of File status occurred.
09	Clear controller failed to clear End of File status.
0A	Clear controller failed to clear Interrupt status.
0B	Clear controller failed to clear End of Operation status.
0C	Clear controller failed to clear separator card status.
0D	Clear controller failed to clear alarm status when alarm status was produced by separator card status.
0E & 0F	Spares.
10	Reload memory function failed. Either the function was not performed when selected or clear controller did not clear its selection.
11	The 1726 card reader test received control from an interrupt, but cannot identify the interrupt.
12	A Busy status occurred when it was not expected, or did not drop.
13	1726 Reader Ready status was not set when it should be.
14	Interrupt status (bit A_2) was not set when a Data Interrupt occurred.
15	Interrupt status (bit A_2) was not set when an End of Operation interrupt occurred.
16	Interrupt status (bit A_2) was not set when an alarm interrupt occurred.

Error Code	Comments
17	Feed failure occurred
18	Stacker full or jam
19	Input tray empty
1A	Preread or compare error
1B	Manual switch in manual position, or motor power off
1C	False End of File status
1D	No alarm interrupt occurred when expected (alarm interrupts properly enabled and alarm status up).
1E	No data interrupt occurred when expected.
1F	No interrupt on End of Operation occurred when expected.
20	The alarm status bit is not set, but some alarm condition is present.
21	Alarm status is present, but no alarm conditions are present.
22	Protect status is present, but an unprotected I/O instruction was not rejected.
23	An unprotected I/O instruction was rejected, but Protect status bit is not set.
24	In checking Protect status, the Clear Controller function was executed, but returned to the External Reject address.
25	False binary status
26	No expected binary status
27	False separator card status
28	No expected separator card status
29	Spare
2A	Binary pattern error - an error occurred in the data read from 25 or 52 cards (4 stops) Typeout and stop meanings: 3rd stop: (A) = actual data (Q) = expected data 4th stop: (A) = column of data failure

Error Code — Comments

2B — Random card error - an error occurred in the data read from random cards. (4 stops)

Typeout and stop meanings:
3rd stop:
(A) = actual data
(Q) = expected data

4th stop:
(A) = column of data failure

2C — Hollerith error - an error occurred in the data read from Hollerith cards. (4 stops)

Typeout and stop meanings:

3rd stop

(A) = actual data
(Q) = expected data

4th stop

(A) = column of data failure

Note: When reading cards in Hollerith mode, two columns are packed into each memory location. In checking the data, first the upper half of the memory word is checked; then the lower half; each with the other half masked off. Expected and acutal data are displayed in the lower 8 bits of A and Q.

Example:

The data read into core from columns 1 and 2 should be:

3031 (hex)

Suppose it was read

2031

The third stop will display (A) = 0020
(Q) = 0030. The fourth stop will display (A) = 0001 (hex)

2D and 2E — Spares

2F — Internal reject when status was being checked

30 — Unexpected external reject on status

Error Code	Comments
31	Internal reject on function
32	Unexpected external reject on function
33	Internal reject on data input
34	Unexpected external reject on data input
35	Internal reject on data output
36	Unexpected external reject on data output
37	Clear Controller function failed to clear a status it should have, or cleared some status it should not have (3 stops) Typeout and stop meanings: 3rd stop: (A) = actual status (Q) = expected status
38	Clear Interrupts function failed to clear a status it should have, or cleared some status it should not have (3 stops) Typeout and stop meanings: 3rd stop: (A) = actual status (Q) = expected status
39	No Data Ready status when expected.
3A	Unable to generate random and data because columns 10 and 12 of the random card gave a different starting random number from that of columns 20 and 22.
3B	Illegal status on Hollerith cards (separator card or End of File) or binary status did not occur on the separator card at the end of the Hollerith section. (3 stops) 3rd stop: (A) = actual status (Q) = expected status for Hollerith cards
3C	No reject occurred when we attempted to do a function while the card reader is Busy.
3D	The 405 is slow in picking cards through. May be caused by maladjustment of the 405 or worn cards.

III. DESCRIPTION

This test checks proper operation of the 1726/405 card reader attached to either the AQ or the buffered channel of the 1704 computer. See ID for a description of the test deck. The test is divided into seven operating sections which may be selected at parameter entry time. Repeat conditions, sections, stops on end of sections, etc., are chosen in the standard SMM17 format. Operations of the test sections are briefly described below: Sections 1 through 6 are executed with alarm interrupts enabled, in order to detect and report any feed failures, stacker full of jam conditions, etc.

Section 0 - Initialization

A. Sense End of File status, Manual status, Separator Card status, and Pre-read/ Compare status at beginning of test, and report if any of these conditions are set initially.

B. Sense Busy and Ready status and report whether the 1726 is Busy or Not Ready.

C. If any alarm conditions are present, after reporting them, enable interrupt on alarm and check that we do get an interrupt properly. Wait until alarm conditions have been removed before going on.

Section 1 - Hollerith cards

A. Read Hollerith cards, checking proper operation of Data Interrupts and Interrupt or End of Operation. Check binary status, Separator Card status, and End of File status. If binary status is not present, but End of File or separator card is, report error. If binary status is present, check to see whether the terminiting separator card has been read, and if not, report false binary status. Check Hollerith data. Check proper operation of clear interrupt/clear controller (the clear interrupt function is done on even cards, the clear controller, on odd cards).

Section 2 - Protect test

A. Attempt to do an unprotected Clear Controller function when the Protect switches are set on the console and on the 1726. Check that the function is rejected. Check presence of Protect status bit.

Section 3 - 25 cards

A. Read 25 cards after having done release negate Hollerith to ASCII function. Check data, thus testing whether the 79 punch in column 1 causes the card to be read as binary data. Check presence of binary status and absence of separator card status.

Section 4 - 52 cards

A. Read 52 cards after having done negate Hollerith to ASCII function. Check presence of separator card and binary status. Check that alarm interrupt occurs on separator card status. Check data.

Section 5 - Separator cards

A. Read separator cards, checking operation of the reload memory function. Every other card is completely read, with alternate cards executing reload memory function after reading one column. Check separator card and alarm status, and whether alarm interrupt occurred. Execute clear controller and be sure separator card, Alarm, Binary, and End of Operation status are cleared by clear controller. Repeat for each card until all have been read.

Section 6 - Random/Gate/End of File

A. Read random cards, checking random data column by column. Random data is generated from the contents of the lower half of columns 10, 12, 20, and 22 of the card. Note that random cards have random densities, ranging from approximately 1/8 of the holes punched on the card to 7/8 of the holes punched.

B. If blank cards are inserted after the random cards, test proper execution of gate card function. Every other blank card should be gated into the secondary stacker.

C. Check End of File status (unless deleted at parameter entry time). The EOF switch should be set on the console throughout the test. Check that EOF status does come up, and that it is cleared by a clear controller function. Also check that a function is rejected when the reader is Busy.

III. PHYSICAL REQUIREMENTS

A. STORAGE REQUIREMENTS - Approximately 6K.

B. TIMING - Variable depending upon the size of test deck.

C. EQUIPMENT CONFIGURATION

1. 17X4 Computer with a minimum of 8K memory.
2. 1726/405 Card Reader.
3. A device for loading the test.

1731/601, 602, 612 MAGNETIC TAPE TEST

(MT1007 Test No. 7)
(CPOF)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

Specific sections will only run on certain types of tape units whether the specific section is selected or not.

1. No sections will run on 7-track tape at 800 BPI.
2. No sections will run on 9-track tape if density is 200 or 556 BPI or if mode is BCD.
3. Bit 15 of Unit Select parameter must be set for BDC direct I/O.

B. LOADING PROCEDURE

This test operates under the control of 1700 SMM. The calling sequence is that specified by SMM. To restart test after loading MC, set P = IA and RUN.

C. PARAMETERS

If Bit 0 of the Stop/Jump word is set, the program will have one monitor stop displaying $741 in the A register and the Stop/Jump word in the Q register, and three program stops to enter parameters in A and Q as follows (after last stop parameters selected will be typed out):

Stop 2	A = Section Selects
	Q = Unit Selects, Bit 15 = Direct BDC I/O

Stop 3	A = Data Interrupt line
	Q = Alarm Interrupt line

Stop 4	A = End of Operation Interrupt line
	Q = Zero

1. Section Select Bits

Bit	Section	Function
0	1	Ready Not Busy status
1	2	Write Enable Load Point, Protect status
2	3	Binary parity and File Mark status
3	4	BCD Parity and File Mark status
4	5	Read wrong mode parity
5	6	Write, read, and check file mark and binary data
6	7	Binary patterns
7	8	BCD data
8	9	Variable length records
9	10	Creep Test
10	11	Ladder Test
11	12	Interrupts

2. Unit Select Bits

Bit	Unit	Function
0	0	Units selected must be Ready and Write Enabled and may be any combination of 601's, 602's, and 612's. No new units may be selected after initial parameter.
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
15	N/A	Set for direct I/O on BDC channel

D. STOP AND JUMP SETTINGS

Bit Settings in STJP Word

Bit	Stops
0	Stop Enter Parameters
1	Stop End of Section (bits 12, 13, and 14 indicate unit number)
2	Stop End of Test
3	Stop on Error

Bit	Jumps
4	Repeat Conditions
5	Repeat Section
6	Repeat Test
7	Omit Simulation
8	Omit Typeouts
9	Bias Return Address Display
10	Re-enter Parameters (changeable parameters are section select bits and direct or buffered BDC I/O)

E. MESSAGES

1. Alarms and Typeouts

a. Normal Program Typeouts

```
MT1007, 1731 MAGNETIC TAPE TEST
IA = XXXX, FC = XX , CPOF, VER. 3.1
```

b. Error Alarm Typeouts

0001 - External Connect reject
0002 - Unit Not Ready
0003 - Status reject
0004 - Unit Busy
0005 - Unit protected
0006 - No write enable
0007 - W = 02, 07, or 0C
0008 - No units available
0009 - External Function reject
000A - No Load Point status
000B - Internal reject on output
000C - Internal reject on input
000D - No Busy status
000E - No File Mark/Tape Mark status
000F - No Parity status
0010 - Parity status
0011 - No 7-track status
0012 - Deselect reject
0013 - Data Compare error
0014 - Internal Connect reject
0015 - No Parity status on binary file mark

0016 - BDC External reject on I/O
0017 - No Interrupt status
0018 - Non-requested interrupt
0019 - Unidentified interrupt
0020 - Alarm interrupt - Not Ready
0021 - Alarm interrupt - Lost Data
0022 - Alarm interrupt - Parity Status
0023 - Alarm interrupt - End of Tape
0024 - Alarm interrupt - File Mark/Tape Mark
0025 - No alarm interrupt condition
0026 - End of Operation interrupt not cleared
0027 - Alarm interrupt not cleared
0028 - Data interrupt not cleared
0029 - Parity status on binary tape mark
0030 - Internal function reject
0031 - Internal Mode/Density Select reject
0032 - Internal Deselect reject
0033 - No alarm status of file mark/tape mark
0034 - Unidentified interrupt not cleared
0035 - No End of Operation interrupt
0036 - No Data interrupt
0037 - I/O reject following data interrupt
00FF - Not sufficient core length

2. Error Displays

There are three types of error displays: data error (ECC$13), core length (ECC$FF), and non-data errors. Displays are as follows:

	A	Q	A	Q	A	Q
Non-data	ID	STJP	SECECC*	RETAD**		
Data	ID	STJP	SECECC*	RETAD**	Actual data	Expected data
					A	
Core Length	ID	STJP	SECECC*	RETAD**	Core Length	Last AD

*SECECC: Section Number - bits 8-15; Error Code - bits 0-7
**RETAD: Return Address biased

F. ERROR STOPS

For all errors there are two monitor stops with displays in A and Q. Bits 4 to 7 of the ID word indicate the total number of stops.

II. DESCRIPTION

This test consists of 12 sections which can be individually selected. All sections will be executed unless the Section Select bits are altered by the operator. Each section selected is run on each unit in consecutive unit number order until all sections have been run on all units. This procedure is repeated for each density.

A. SECTION MT0

Ready Not Busy Status

1. Check Ready status.
2. Check Repeat conditions.
3. Check Busy status.
4. Check Repeat conditions.
5. Repeat 500_{10} times beginning with step 1.

B. SECTION MT1

Write Enable, Load Point and Protected Status

1. Rewind unit.
2. Check Write Enable status.
3. Check Repeat conditions.
4. Check Load Point status.
5. Check Repeat conditions.
6. Check Protected status.
7. Check Repeat conditions.
8. Repeat 500_{10} times beginning with step 2.

C. SECTION MT2

Busy on BDC Write, Parity and File Mark/Tape Mark Status

1. Check Channel.
   a. BDC channel - continue with step 2
   b. AQ channel - jump to step 5
2. Write 300_{10} word record to all "1's".
3. Check Busy status of the BCD 10 times.

4. Wait for Not Busy.
5. Write File Mark/Tape Mark.
6. Wait for End of Operation status.
7. Check track number of unit.
    a. If 7 track - check for presence of parity status
    b. If 9 track - check for absence of parity status
8. Check Repeat conditions.
9. Check File Mark/Tape Mark status.
10. Check Repeat conditions.
11. Check Alarm status.
12. Repeat 10 times from step 5.

D. SECTION MT3

Write BCD File Mark, Check Busy and File Mark Status

1. Write File Mark.
2. Checks for Busy.
3. Check Repeat conditions.
4. Wait for Not Busy.
5. Check File Mark status.
6. Check Repeat conditions.
7. Repeat 10 times from step 1.

E. SECTION MT4

Parity Status - Write/Read Opposite Mode

1. Write two 125_{10} word binary records.
2. Write two 125_{10} word BCD records.
3. Backspace four times.
4. Read first binary record as BCD.
5. Wait for Not Busy.
6. Check for expected parity status.

7. Read second binary record as binary.
8. Wait for Not Busy.
9. Check for absence of parity status.
10. Read first BCD record as binary.
11. Wait for Not Busy.
12. Check for expected parity status.
13. Read second BCD record as BCD.
14. Wait for Not Busy.
15. Check for Absence of parity status.
16. Check 7-track status.
17. Check repeat conditions.
18. Repeat 100_{10} times from step 16.

F. SECTION MT5

Write, Read, and Check Binary Record and File Mark/Tape Mark data

1. Rewind unit.
2. Write six file marks/tape marks.
3. Write a 40-word binary record of $FF.
4. Write six file marks/tape marks.
5. Backspace seven times.
6. Read and check 40-word binary record.
7. Backspace once.
8. Backspace once.
9. Wait for Not Busy.
10. Check File Mark/Tape Mark status.
11. Repeat six times from step 8.
12. Backspace once.
13. Wait for Not Busy.
14. Check Load Point status.
15. Read and check one file mark/tape mark.
16. Repeat step 15 six times.

G. SECTION MT6

Write, Read, and Check Five Binary Patterns

1. Patterns used.

   a. all "0's"

   b. all "1's"

   c. $AA55

   d. left-shifting 0

   e. left-shifting 1

2. Write five 100_{10} word record of pattern.

3. Backspace five times.

4. Read and check data of the five 100_{10} word record.

5. Repeat from step 2 until all patterns have been written, read and checked.

H. SECTION MT7

Write, Read, and Check BCD Records

1. Write file mark.

2. Wait for Not Busy.

3. Generate 100_{10} BCD codes.

4. Write five 100_{10} word BCD record.

5. Backspace five times.

6. Read and check data five 100_{10} word BCD records.

7. Repeat from step 1 five times.

I. SECTION MT8

Variable Length Records of $20

1. Write file mark/tape mark.

2. Wait for Not Busy.

3. Write six binary records of length 28_{10}, 97_{10}, 275_{10}, 20_{10}, 75_{10}, and 5_{10}.

4. Backspace six times.

5. Wait for Not Busy.

6. Read and check each record.

7. Repeat from step 3 five times

J. SECTION MT9

Creep

1. Write 20_{10} file marks/tape marks.
2. Write a one word binary record of $F0.
3. Backspace.
4. Read and check one word record.
5. Repeat from step 2 fifty times.

K. SECTION 10

Ladder Test

1. Write file mark/tape mark.
2. Write $FF binary record of variable length and comparable data.

Length (words)	Data
1	$01
2	$02
3	$03
.	.
.	.
$20	$20
.	.
.	.
$FF	$FF

3. Backspace $FF times.
4. Read and check each binary record.

L. SECTION MT11

Interrupts - Alarm interrupt is selected throughout the entire test.

1. Select End of Operation interrupt.
2. Set End of Operation interrupt request flag.
3. Write a 500_{10} word record of one's.
4. Wait for Not Busy .
5. Check End of Operation occur flag.

6. Clear End of Operation occur flag.

7. Reset End of Operation interrupt.

8. Backspace once.

9. Wait for Not Busy.

10. Check End of Operation occur flag.

11. Clear End of Operation occur flag.

12. Read 500_{10} word record that was written.

13. Wait for Not Busy.

14. Check End of Operation occur flag.

15. Clear End of Operation occur flag.

16. Clear all interrupts.

17. Clear End of Operation Request flag.

18. Check data.

19. Repeat from step 1 ten times.

20. Check channel.

    a. BDC - set Alarm Interrupt Request flag and exit from section.

    b. AQ - check data interrupts as follows:

       1) select Data interrupt

       2) set Data Interrupt request flag

       3) select Write function

       4) wait for Not Busy

       5) check data occur flag

       6) clear Data occur flag

       7) check repeat conditions

       8) repeat from step 1 selecting a Read function

    c. Reset Read function back to Write.

    d. Store 25_{10} words in output area.

    e. Clear interrupts.

    f. Clear data line from Mask register.

g. Select Data interrupt.

h. Select Write function.

i. Check Interrupt status bit.

j. Write one word.

k. Repeat from step i 25 times.

l. Backspace.

m. Wait for Not Busy.

n. Repeat from step e to step k changing Write's to Read's.

o. Set data line back in Mask register.

p. Reset Alarm Request flag.

III. INITIALIZATION AND SUBROUTINE DESCRIPTION

A. INITIALIZATION

1. Convert bias value and frequency count to ASCII and store in typeout routine.
2. Check core length.
3. Set all section select bits.
4. Set up title for typeout.
5. Set up interrupt line display for parameter stop.
6. Check W for zero or correct value for BDC channel.
7. Store unit selected by operator or units Ready and Write Enabled in a table.
8. Clear controller.
9. Store pattern for Section MT6.
10. Store pattern lengths for Section MT8.
11. Store density for pass 1 of test.

B. SUBROUTINES

1. TESTEX - Intermediate exit to monitor

   Stores return address of test at IA + 5

2. WNB - Wait for Not Busy

   a. Check BDC for Not Busy if on BDC channel

   b. Obtain status 1

   c. Check Busy status

   d. If Not Busy, exit

   e. If Busy, store direct locations used both by monitor and test such as I and $10 to $19

   f. Exit through TESTEX

   g. Restore values stored at step e.

   h. Repeat from step until Not Busy

3. CON - Connect

   a. Connect unit

   b. Check Ready Not Busy, Protected, and Write Enable status

   c. Determine whether particular unit connected will run at specified density and mode

   d. If no, next unit in table is connected

   e. If yes, mode and density (MDEN) are selected

4. SECT - Check Section Select Bit

   a. Section bit is stored

   b. Mode and density are stored

   c. Determine whether section is selected

   d. If yes, add one to return address and do an RTJ to CON and exit SECT

   e. If no, exit SECT

5. DATA - Generate and Compare Data

   a. Store number of words and pattern

   b. Generate data output one word at a time and compare it with that input

6. STOP - Routine Used for Error and End of Section Stops

   a. Set up stop display

   b. Unit number is bits 12-14 of ID word

7. OUT - Write Data on AQ or BDC Direct or Buffered

   a. Generate and store all data to be written except for section 7 which generates its own data

   b. Write and generate data in the manner requested by W and bit 15 of unit parameter.

   c. BDC present - do an RTJ to DS1

   d. RTJ - WNB

8. INP - Read Data on AQ or BDC Direct or Buffered

   a. Type of input is determined by W and bit 15 of units parameter

   b. RTJ - WNB

9. SLFUN - Select All Controller Functions

   a. Check status for End of Operation

   b. If not present, do an RTJ to WNB

   c. If present, select requested function

10. MDEN - Select Mode and Density

   a. Loop on controller active

   b. Controller inactive, select mode and density requested

11. DS1 - Obtain Status 1

   a. Check BDC for Not Busy if on BDC channel

   b. Store status of 1731 at STAT1.

12. DS2 - Obtain Status 2

   a. Check BDC for Not Busy if on BDC channel

   b. Store status of 1731 at STAT2

13. ENDSC - Check Repeat Section

   a. Set A and Q for End of Section stop

   b. Check repeat section bit

   c. Deselect units

d. Loop on controller active

e. Update address of units table

14. PARM - Re-enter Parameters

a. Reinitialize address of units table

b. Check Re-enter Parameter bit

c. If set, restore Section Select bits

15. FINAL - End of Test

a. Make sure all densities have been run

b. Display End of Test information

c. Check Repeat Test bit

IV. PHYSICAL REQUIREMENTS

A. SPACE REQUIREMENTS - 2438_{10} or 986_{16} locations

B. TEMPORARY STORAGE REQUIREMENTS - 500_{10} or $1F4_{16}$ locations

C. TIMING - 3 min 15 sec with one unit

D. EQUIPMENT CONFIGURATION

1. 17X4 Computer with 8K memory
2. 1705 Interrupt Data Channel
3. 1706 Buffer Data Channel (optional)
4. 1731 Magnetic Tape Controller
5. Minimum of one 601, 602, or 612 tape unit and a maximum of eight such units in any combination.

1731/601, 602, 612 MAGNETIC TAPE TEST

(MT200E Test No. 0E)
(CPOF)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

None

B. LOADING PROCEDURE

This test operates under the control of the 1700 SMM. The calling sequence is that specified by SMM. Test can be restarted at initial address.

C. PARAMETERS

The initialization portion of the test sets parameters to test all 601's, 602's, and/or 612's connected to the 1731 that are Ready and have file protect rings inserted. It also sets the interrupt line(s) parameters. If a parameter stop is selected, these parameters may be altered. A typeout of parameters will occur after last stop.

1. Selective Stops

a. Parameter	Stop 1:	A = 0E41 OVERFLOW light on
		Q = Stop Jump parameter

	Stop 2:	A = Selected units
		Q = Selected test sections

	Stop 3:	A = Data interrupt line
		Q = Alarm interrupt line

	Stop 4:	A = End of Operation interrupt line
		Q = Units that are 602's

b. End of Section	Stop 1:	A = 0E22 OVERFLOW light on
		Q = Stop Jump parameter

	Stop 2:	A = 0X00 (x = section number)
		Q = Return address

c. End of Test	Stop 1:	A = 0E24 OVERFLOW lite on
		Q = Stop Jump parameter

	Stop 2:	A = Pass count
		Q = Return address

d. Error (See Section E)

2. Selecting Stops and Jumps (Stop Jump Word)

Bit	Stops
0	Stop to enter parameters
1	Stop at end of section
2	Stop at end of test
3	Stop on error

Bit	Jumps
4	Repeat identical conditions of error
5	Repeat section
6	Repeat test
7	Omit simulation
8	Omit typeouts
9	Bias return address display

D. MESSAGES

1. Typeouts

a. Normal Program Typeouts

1) Initialization of test

```
MT200E, 1731 MAGNETIC TAPE TEST
IA = XXXX, FC = XX, CPOF, VER. 3.1
```

2) End of one pass through test

A	Q	A	Q
0E24	Stop/Jump	Pass Count	Return Address

2. Error Types

a. Sense Test Errors (Section 0)

00 - Unit should be Ready at start
01 - Unit does not have write enable
02 - Unit should be Not Busy
03 - Unit should be in 800 BPI
04 - Unit should not be in 556 BPI
05 - Unit should be in 556 BPI
06 - Unit should not be in 800 BPI
07 - Unit should be in 200 BPI

08	-	Should be at load point
09	-	False End of Tape
0A	-	False End of Operation
0B	-	Should sense Busy
0C	-	Should not sense Parity Error, Alarm, Load Point, End of File, Lost Data
0D	-	Should sense controller not active
0E	-	Should sense controller active
0F	-	Should sense Parity Error alarm
10	-	Should sense Parity Error
11	-	Should sense Lost Data
12	-	Should sense Lost Data alarm
13	-	Should sense End of File mark
14	-	Should sense End of File Mark alarm
15	-	Illegal reply after Lost Data condition
16	-	Should not sense Lost Data

b. All others

01	Write parity error
02	Read parity error
03	Data error
04	Block too short
05	Lost data
06	End of Operation interrupt error
07	Data interrupt error
08	Alarm interrupt error
7E	Internal reject
7F	Illegal external reject

E. ERROR STOPS AND TYPEOUTS

A	Q	A	Q	A	Q
NEX8	Stop/Jump	0YZZ	Return Address	VVVV	WWWW

Where N = tape unit number

X = number of stops

Y = section (always zero for reject error)

ZZ = error type

VVVV = 1) failing data word for data error
2) (A) when illegal reject occurred

WWWW = 1) expected data word for data error
2) (Q) when illegal reject occurred

II. DESCRIPTION

A. The program consists of seven sections. It tests all units being used simultaneously. Each section is run in all recording densities.

1. Section 0 - Sense

   Checks all sensing - both for the condition and for absence of condition where applicable. Repeat five times.

2. Section 1 - Interrupt

   a. Write file mark. Check End of Operation interrupt

   b. Read file mark. Check Alarm interrupt

   c. Check Write Data interrupt

   d. Check Read Data interrupt

   e. Repeat section in all densities

3. Section 2 - Backspace

   a. Write 18_{10}-word binary record of all zeros

   b. Write eight file/tape marks

   c. Backspace nine times

   d. Read all "0's" record

   e. Write 18_{10}-word binary record of all "1's"

   f. Write 50_{10} 18_{10}-word binary records of all "0's"

   g. Backspace 51_{10} times

   h. Read all "1's" record

   i. Repeat section in all densities

4. Section 3 - Long Record

   a. Write 4000_{16}-word binary record

   b. Backspace

   c. Read 4000_{16}-word record (no data checking)

   d. Repeat section in all densities

5. Section 4 - Variable Record Length - BCD

   a. Write BCD pattern in records ranging from 18_{10} words to 258_{10} words in increments of four

b. Backspace to first record

c. Read all records

d. Repeat section in all densities

6. Section 5 - Random Block Length

   a. Generate random data in write data area

   b. Generate random record length

   c. Write 64_{10} random length records

   d. Backspace 64_{10} records

   e. Read 64_{10} records

   f. Repeat section in all densities

7. Section 6 - Nonstop Read and Write

   a. Write 80_{10} records in Start/Stop mode

   b. Backspace 80_{10} records nonstop

   c. Read 80_{10} records nonstop

   d. Write 80_{10} records nonstop

   e. Backspace 80_{10} records start/stop

   f. Read 80_{10} records start/stop

   g. Repeat section in all densities

B. INITIALIZATION AND SUBROUTINE DESCRIPTION

1. Initialization

   a. Convert bias value and frequency count to ASCII and store in typeout routine.

   b. Store return address.

   c. Find converter number for BCD operation to determine which Busy switch to use, if applicable.

   d. Find all units that are ready to write.

   e. Find interrupt lines to 1731.

   f. Type test title, number, initial address, and frequency count.

2. Subroutines

   a. BSYCON - Passes control to and accepts control from SMM17. Checks and sets or clears the converter Busy switch if applicable (1706, 1716)

   b. SUNO - Select Unit

      1) Selects next available unit

      2) Upon entry A = parity mode parameter. If A = 0, use previous parameter

      3) Upon exit A = 0 if another unit is selected. A ≠ 0 if no more units.

      4) Checks SELECTIVE SKIP switch.

   c. ILREJ - Reject Error

      1) Stores A and Q in display

      2) Go to SMM Stop subroutine

   d. STATO - Level 1 Status

      1) Upon entry if A = 0, read status and exit

      2) Upon entry if A ≠ 0, wait for condition in A and give control to SMM17 until condition is present

      3) Upon entry A can be either positive or negative when not zero

   e. LEV2 - Level 2 Status

      1) Read status and exit

   f. FSELO - Perform function in A to selected unit

   g. FALLO - Perform function in A to all units

   h. WSELO - Write on selected unit

      1) A contains record length

      2) Control given to SMM17 after buffer initiate

   i. WALLO - Write one record on all units

      1) A contains record length

      2) Check for errors

      3) Uses WSELO

j. RSELO - Read on selected unit

1) A contains record length, if known

2) If A = 0, record length will be determined and recorded

3) Uses End of Operation interrupt

4) Gives control to SMM17 after buffer initiate

k. RALLO - Read one record all units

1) Uses RSELO

2) Checks all errors

l. DENO - Update density select parameter

m. PATTO - Stores A in 80_{10} consecutive buffer locations

n. TOERR - Section 0 error

Processes all errors detected in section 0

III. PHYSICAL REQUIREMENTS

A. TEMPORARY STORAGE REQUIREMENTS

After all tests called by SMM have been loaded and initialized, the program will attempt to reserve 1024_{10} locations. If not available, it will use any amount less than 1024_{10}.

B. TIMING - 2 min 30 sec with one unit.

C. EQUIPMENT CONFIGURATION

1. 1704 Computer with 4K memory
2. 1705 Interrupt Data Channel
3. 1706 Buffer Data Channel (optional)
4. 1731 Magnetic Tape Controller
5. Minimum of one 601, 602, or 612 Tape Unit and a maximum of eight such units in any combination.

1732/608, 609 - 1732-2/615-73/93 MAGNETIC TAPE TRANSPORT TEST

(MT3015 Test No. 15)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

None

B. LOADING PROCEDURE

This test operates under control of SMM17. The calling sequence is that specified by SMM17. This test may be restarted at initial address. If equipment address FCN when test is loaded = 3, this will preselect 1732-2 DSA I/O.

C. TESTING PROCEDURE

If 1732-2, the whole test will run either in A/Q or DSA mode but <u>not</u> both at the same time.

D. PARAMETERS

The initialization portion of the test sets parameters to test all 608's and 609's connected to the 1732 that are ready and have file protect rings inserted. The test will run buffered if a 1706/16 channel is available. A typeout of parameters will occur after last stop.

1. <u>Selective Stops</u>

   a. If a parameter stop is selected, these parameters may be altered;

   Stop 1: A = 1541 OVERFLOW light on
   Q = Stop Jump Parameter

   Stop 2: A = Selected units (e. g. bit 5, unit 5)
   Q = Selected sections (e. g. bit 6, section 6)

   Stop 3: A = Data Interrupt Line
   Q = Alarm Interrupt line

   Stop 4: A = End of Operation Interrupt Line
   Q = 0000 = 1732 or 1732-2 A/Q
   ≠ 0000 1732-2 DSA operations only

   b. End of Section

   Stop 1: A = 1522 OVERFLOW light on
   Q = Stop Jump Parameter

   Stop 2: A = 0X00 (X = section number)
   Q = Return Address

c. End of Test

Stop 1:	A = 1524 OVERFLOW light on
	Q = Stop/Jump Parameter

Stop 2:	A = Pass count
	Q = Return Address

d. Error (see section E)

2. Selecting Stops and Jumps (Stop Jump Word)

Bit	Stops
0	Stop to enter parameters
1	Stop at end of section
2	Stop at end of test
3	Stop on error

Bit	Jumps
4	Repeat identical conditions of error
5	Repeat section
6	Repeat test
7	
8	Omit typeouts
9	Bias return address display
10	Not used (Use bit 0 instead)

D. MESSAGES

1. Typeouts

a. Normal Program Typeouts

1) Initialization of Test

```
MT3015, 1732-2 MAGNETIC TAPE TEST
IA = XXXX, FC = XX  CP = 2F  VER. 3.1
```

2) End of pass X through test

A	Q	A	Q
1524	Stop/Jump	X	Return Address

2. Error Types

a. Sense Test Error (Section 0)

00 - Ready status should be set

01 - Write enable status not set

02 - Busy status set (should not be set)

03 - 800 BPI status not set (should be set)

04 - 556 BPI status is set (should not be set)

05 - 556 BPI status not set (should be set)

06 - 800 BPI status is set (should not be set)

07 - 200 BPI status is not set (should be set)

08 - Load point status is not set (should be set)

09 - End of tape status set (should not be set)

0A - End of Operation status set (should not be set)

0B - Busy status is not set (should be set)

0C - One or more of the following status bits are set; Parity error, Alarm, Load point, End of file, Lost data (None of these should be set)

0D - Controller Active status is set (should not be set)

0E - Controller Active status is not set (should be set)

0F - Alarm status is not set (should be set from parity error)

10 - Parity Error status is not set (should be set)

11 - Lost Data status is not set (should be set)

12 - Alarm status is not set (should be set from forcing lost data)

13 - File Mark status not set (should be set)

14 - Alarm status is not set (should be set from reading file mark)

15 - Attempted to input data when Data status was not present. Input to A instruction should have rejected, but it did not.

16 - Lost Data status is set (should not be set)

17 - Fill status is not set (should be set)

b. All other Errors

01 - Parity Error on Write operation

02 - Parity Error on Read operation

03 - Data compare error (reports only the first data error in a record)

04 - Block too short

05 - Lost Data

06 - End of Operation Interrupt error

07 - Data Interrupt error

08 - Alarm Interrupt error

09 - Fill status set (should not be set)
69 - Controller stayed busy too long; possibly caused by a long rewind
7E - Internal Reject

7F - Illegal External Reject

E. ERROR STOPS AND TYPEOUTS

A	Q	A	Q	A	Q	A	Q
15X8	Stop/Jump	NYZZ	Return Address	VVVV	WWWW	WDCNT	RL

Where X = number of stops
N = unit number
Y = section number
ZZ = error type
VVVV = 1) actual data on data compare error
2) (A) when illegal reject occurred
WWWW = 1) expected data on data compare error
2) (Q) when illegal reject occurred
WC=(A) word number in error (data compare error only)
RL= (Q) expected record length of data error

II. DESCRIPTION

A. NINE SECTION PROGRAM (NUMBERED 0-8)

All units being used are tested simultaneously. Each section is run in all recording densities.

1. Section 0 - Status Sense Test

This section checks all status bits - both for the condition and for absence of the condition where applicable. Repeat 5 times.

2. Section 1 - Interrupt Test

a. Rewind all units. Write file mark. Check End of Operation interrupt.

b. Read file mark. Check Alarm interrupt

c. Check Write Data interrupt

d. Check Read Data interrupt

e. Repeat section in all densities

3. Section 2 - Backspace Test

a. Rewind all units. Write 18_{10} - word binary record of all zeros

b. Write eight file/tape marks

c. Backspace nine times

d. Read all zeros record

e. Write 18_{10} - word binary record of all ones

f. Write 50_{10} - 18_{10} - word binary records of all zeros

g. Backspace 51_{10} times

h. Read all ones record

i. Repeat section in all densities

4. Section 3 - Search File Mark Forward/Backward Test

a. Rewind all units

b. Write file mark on all units

c. Write 18_{10} - word record of 0000_{16} on all units

d. Write file mark on all units

e. Write 18_{10} - word record of 0001_{16} on all units

f. Repeat from step d (incrementing the data of each record by 0001_{16}) until a record of 0030_{16} has been written.

g. Rewind all units

h. Execute 19_{16} search file mark forward functions on all units.

i. Read record of 0018_{16}'s on all units

j. Execute 14_{16} search file mark forward function on all units.

k. Execute 19_{16} search file mark backward functions on all units

l. Execute 1 search file mark forward function on all units

m. Read record of 0014_{16} on all units

n. Repeat from Step a for all densities

5. Section 4 - Variable Record Length - BCD (Binary if 609)

   a. Rewind all units. Write BCD pattern in records ranging from 18_{10} words to 258_{10}words in increments of four (character mode)

   b. Backspace to first record

   c. Read all records (character mode)

   d. Repeat section in all densities

6. Section 5 - Random Block Length - Binary

   a. Rewind all units. Generate random data in write data area

   b. Generate random record length

   c. Write 64_{10} random length records (character mode)

   d. Backspace 64_{10} records

   e. Read 64_{10} records (character mode)

   f. Repeat section in all densities

7. Section 6 - Nonstop Read and Write

   a. Rewind all units. Write 80_{10}records in Start/Stop mode

   b. Backspace 80_{10} records nonstop

   c. Read 80_{10} records nonstop

   d. Write 80_{10} records nonstop

   e. Backspace 80_{10} records start/stop

   f. Read 80_{10} records start/stop

   g. Repeat section in all densities

8. Section 7 - Variable Record Length - BCD (Binary if 609)

   This section is identical to Section 4 except that all data is written and read in assembly/disassembly mode rather than character mode.

9. Section 8 - Random Block Length - Binary

   This section is identical to Section 5 except that all data is written and read in assembly/disassembly mode rather than character mode.

B. INITIALIZATION

1. Convert bias value and frequency count and store in typeout routine.
2. Store return address.
3. Find converter number for BAQ operation to determine which BUSY switch to use, if applicable.
4. Finds all units that are Ready and Write enabled.
5. Types test title, number initial address, and frequency count.

III. PHYSICAL REQUIREMENTS

A. TEMPORARY STORAGE REQUIREMENTS

After all tests called by SMM have been loaded and initialized, the program will attempt to reserve 1024_{10} locations. If not available, it will use any amount less than 1024_{10}.

B. EQUIPMENT CONFIGURATION

1. 17X4 Computer
2. 1732 Magnetic Tape Controller or 1732-2
3. Minimum of one 608 or 609 Magnetic Tape Transport and maximum of eight of any combination of such units, or 615-73/93 NRZI only
4. 1706 or 1716 Buffered Channel (optional)

1731/1732, 601, 608, 609 SPECIAL MAG TAPE TEST
(MTSA1F Test No. 1F)
(C P0F)

I. OPERATIONAL PROCEDURE

A. RESTRICTIONS

1. This test must be run alone.
2. Bit 5 in the SMM parameter word must be set. This tells the monitor to use the Non-interrupt mode typeout.
3. After loading test, set all unit PROTECT switches on tape controller, and if controller is 1732, set TEST MODE switch on tape controller. All units that have Write enabled and are Ready will be tested.
4. Check special instructions under Test Description, II. D, Section 3, before running test.
5. Sections are run in the following order: 1, 5, 2, 3, 4.

B. LOADING PROCEDURE

The calling sequence is that specified by SMM17. The test number for this special mag tape test is 1F. Test can be restarted at initial address.

C. PARAMETERS

1. If bit 0 of the Stop/Jump word is set, a parameter stop will occur.

   a. First stop, A = 1F31, Q = Stop/Jump parameter. The Stop/Jump parameter may be changed if desired.

   b. Second stop, A = controller designator and Section selection:

      Bit 15 = 1: Controller is a 1732
      Bit 15 = 0: Controller is a 1731
      Bit 04 = 1: Run test Section 5
      Bit 03 = 1: Run test Section 4
      Bit 02 = 1: Run test Section 3
      Bit 01 = 1: Run test Section 2
      Bit 00 = 1: Run test Section 1

      Q = Data interrupt line. The bit position identifies the interrupt line, i. e., bit 6 = interrupt line 6.

c. Third stop, A = Alarm interrupt line, Q = EOP interrupt line. The bit position identifies the interrupt line.

2. Prestored parameters as follows:

a. Controller is a 1731, sections 1 through 4. If no sections are selected, the End of Test stop will indicate the units that are Ready and Write enabled.

b. Data interrupt is on line 2.

c. Alarm interrupt is on line 3.

d. End of Operation interrupt is on line 4.

3. A typeout of selected parameters will occur after last stop.

D. MESSAGES

1. Typeouts

a. Normal program typeout

1) Test identification at start of test
MTSA1F, 1731/32 Special Test
IA = XXXX, FC = XX, CPOF, Ver. 3.1

2) End of Test typeout

1st Stop	A = 1F24	Q = STJP Word
2nd Stop	A = Pass Count	Q = Units tested

Units tested: Bits 00-07 correspond to units 0-7.

b. Error typeouts

1) Error messages are in format specified by SMM17.

A	Q	A	Q
1F38	S/J Parameter	YYZZ	Return Address

YY = Section Number
ZZ = Error Code

2) Additional test information for all errors (third stop):

A	Q
Current Unit	Units Tested

E. ERROR CODES

An error code is displayed in the lower two digits of the A register on the second stop of all error stop sequences.

Error Code	Description
01	No units are Ready and Write enabled.
02	Internal reject on Connect function.
03	Reject on status 2 input.
04	Reject on status 1 input.
05	Incorrect number of words read during CRC check.
06	Incorrect CRC word.
07	Alarm interrupt failed to occur.
08	Incorrect interrupt occurred, received alarm interrupt.
09	Incorrect status after Alarm interrupt.
0A	End of Operation interrupt failed to occur.
0B	Incorrect interrupt occurred, received EOP interrupt.
0C	Incorrect status after EOP interrupt.
0D	Reply received when attempting selection of 9 track and BCD mode, expected Reject.
0E	No parity error status after binary Write and BCD Read.
0F	Reply received from unprotected instruction, expected Reject.
10	Computer PROTECT or tape PROTECT switch was never set, function never rejected.
11	No Load Point status after Rewind.
12	Unexpected EOP or Interrupt status received after a no word Write.
13	Rewind function accepted while unit was Busy.
14	Unexpected EOP status after a no word Write.
15	No load point status after rewind.
16	Backspace function rejected when at load point.

Error Code	Description
17	Unit did not go Busy after backspace from load point.
18	Not used.
19	No load point status after Search File Mark Backward.
1A	Rewind unload function rejected.
1B	Data interrupt failed to occur.
1C	Incorrect interrupt occurred, received data interrupt.
1D	Incorrect status after Data interrupt.
1E	No Protect status after selecting unit.
1F	Unidentified interrupt.

II. TEST DESCRIPTION

A. INITIALIZATION

1. Typeout title.
2. Parameter stop if bit 0 of Stop/Jump word set.
3. Return control to monitor.

B. SECTION 1 - CRC CHECK

1. If 7-track unit selected, test proceeds to next unit.
2. If 9-track unit selected, a binary count is written from $00FF_{16}$ to 0000.
3. Backspaces and reads 0101_{16} words, last word is CRC word.
4. Check is made for correct CRC word.

C. SECTION 2 - CONTROLLER PARITY ERROR CHECK

1. If 7-track unit selected, a word is written in binary mode.
2. Word is read back in BCD mode.
3. Parity error is expected.
4. If 9-track unit selected, an attempt is made to select BCD mode.
5. A Reject is expected.

D. SECTION 3 - CONTROLLER PROTECT CIRCUITRY CHECK

1. The computer will stop with the unit being tested in A, and either "1731" or "1732" in Q. All units will be run which are Ready and have Write enabled.
2. The computer PROTECT switch should be set and run.
3. Momentarily clear unit PROTECT switch on tape controller for unit which was indicated in "A".
4. After resetting unit PROTECT switch, clear PROTECT switch on computer.
5. Computer will stop with next unit to test indicated in "A" and go to 1 or
6. Test will advance to next section when all units have been tested.
7. If computer stops with Protect fault, clear computer PROTECT switch and run. Error message will be displayed.

E. SECTION 4 - EOT

1. An EOT marker must be on tape for this section.
2. Check for Load Point status set after rewind.
3. Check that EOP status does not set after a Write has been initiated, unit going Busy, with no data written.
4. Check that Rewind is not accepted while controller is Busy.
5. Check for proper End of Tape status.
6. Check that backspace from load point is accepted.
7. Check Rewind unload function.
8. In addition, for the 1732 only, a check is made for Search File Backward function from load point.

F. SECTION 5 - CONTROLLER INTERRUPT AND INTERRUPT STATUS

1. Write File Mark function is executed.
2. Check for Alarm interrupt occurring before EOP interrupt.
3. Check for EOP interrupt occurring after the Alarm interrupt.
4. Data interrupt is checked after a write function has been issued.

III. PHYSICAL REQUIREMENTS

A. SPACE REQUIRED: About 1105_{10} locations.

B. EQUIPMENT CONFIGURATION

1. 17X4 computer with 4K memory.
2. 17X5 Interrupt Data Channel.
3. 1706 Buffer Data Channel (Optional)
4. 1731 or 1732 Mag Tape Controllers and 601, 608, or 609 Mag Tape Units.
5. A device for loading program.

COMMENT SHEET

MANUAL TITLE CONTROL DATA® 1700 SYSTEM MAINTENANCE MONITOR (SMM17

Volume 1 Reference Manual

PUBLICATION NO. 60182000 REVISION L

FROM: NAME:

BUSINESS ADDRESS:

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

CUT ALONG LINE

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
Technical Publications Department
4201 North Lexington Avenue
Arden Hills, Minnesota 55112

CUT ALONG LINE

FOLD

FOLD