

Floppy drives & Controller

PRODUCT CODE: MAINDEC-08-DIRXA-D-D
PRODUCT NAME: RX8/RX01 DIAGNOSTIC PROGRAM
DATE: JUNE, 1977
MAINTAINER: DIAGNOSTIC ENGINEERING

COPYRIGHT (C) 1975, 1977
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE LICENSE TERMS. TITLE TO AND OWNERSHIP OF THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.

1. ABSTRACT
2. REQUIREMENTS
 - 2.1 EQUIPMENT
 - 2.2 STORAGE
 - 2.3 PRELIMINARY DIAGNOSTIC PROGRAMS
3. LOADING PRODEEDURE
4. STARTING PROCEDURE
 - 4.1 STARTING ADDRESS LOCATIONS
 - 4.2 OPERATOR ACTION
 - 4.3 PROGRAM ACTION
 - 4.4 TEST PARAMETER SELECTIONS
 - 4.4.1 RESTRICTIONS
5. OPERATING PROCEDURE
6. ERROR DETECTION
 - 6.1 PROGRAM DEFINITIONS OF:
 - 6.1.1 WRITE ERROR
 - 6.1.2 READ ERROR (CRC)
 - 6.1.3 CRC AND DATA ERROR
 - 6.1.4 NO CRC BUT DATA ERROR
 - 6.1.5 CRC BUT NO DATA ERROR
 - 6.1.5.1 SUMCHECK ERROR
 - 6.1.6 SEEK ERROR
 - 6.1.7 PARITY ERROR
 - 6.2 DEFINITIVE ERROR CODES
 - 6.3 UNEXPECTED OR MISSING ERROR CONDITIONS:
 - 6.3.1 MISSING D.D. MARK
 - 6.3.2 UNEXPECTED D.D. MARK
 - 6.3.3 MISSING ERROR FLAG
 - 6.3.4 UNEXPECTED RX01 IRQ
 - 6.3.5 DEVICE TEST HUNG
 - 6.4 SYSTEM FAILURES: (UNKNOWN IRQ)

7. ERROR REPORTING

7.1 EXAMPLES

- 7.1.1 EXAMPLES OF INTERFACE / CONTROL RELATED ERRORS
- 7.1.2 EXAMPLES OF DRIVE / DATA RELATED ERRORS

- 7.1.2.1 SEEK ERROR
- 7.1.2.2 WRITE-CRC AND DATA ERROR
- 7.1.2.3 READ-CRC AND DATA ERROR
- 7.1.2.4 WRITE-CRC BUT NO DATA ERROR

7.2 ERROR RECOVERY

7.3 FATAL ERROR RECOVERY

8. MANUAL INTERVENTION

8.1 FIELD ENGINEERING TROUBLE SHOOTING AIDS

9. DIAGNOSTIC TEST STRATEGY

10. CONSOLE PACKAGE ADDENDUM

11. APT-8 INTERFACES

12. PROGRAM LISTING

1. ABSTRACT
//////////

THIS PROGRAM IS DESIGNED TO TEST THE RX01 FLEXIBLE DISKETTE SUBSYSTEM CONFIGURED AS A COMPLETE SUBSYSTEM, (RX8 INTERFACE, RX01 CONTROL, AND DISKETTE DRIVES), OR AS A PARTIAL SUBSYSTEM (INTERFACE, AND/OR CONTROL). OBVIOUSLY THE MAXIMUM RESOLUTION POSSIBLE IS ACHIEVED BY TESTING THE RX01 DISKETTE SUBSYSTEM CONFIGURED AS A COMPLETE SUBSYSTEM.

REVISION C

THE ADDITION OF THE CONSOLE PACKAGE.
THE REMOVAL OF TEST THREE (3).
SEE LISTING LOCATION FOR EXPLANATION
THE REMOVAL OF DECIMAL CONVERSION ROUTINES.
THE ADDITION OF APT-8 INTERFACES.

REVISION D

THE ADDITION OF THE NEW CONSOLE PACKAGE.
COMPATABILITY FOR VT78 SYSTEMS.
ABILITY TO TEST 2 RX01 UNITS(4 DRIVES).
INTERLEAVING WAS MODIFIED FOR VT78.

2. REQUIREMENTS
//////////
2.1 EQUIPMENT
//////////

A PDP-8/E TYPE GENERATION COMPUTER WITH 4K OF CORE, CONSOLE SWITCHES, TELETYPE, AND AN RX01 DISKETTE SUBSYSTEM.
THE DIAGNOSTIC CAN RUN WITHOUT HARDWARE SWITCH IF THE CONSOLE PACKAGE IS USED. IF THE CONSOLE PACKAGE IS ACTIVE 8K OF CORE IS NEEDED.

2.2 STORAGE
//////////

THIS PROGRAM IS DESIGNED TO BE RUN STAND ALONE AND MUST OCCUPY PROGRAM LOCATIONS 0 THRU 7577 OF FIELD 0.
THE CONSOLE PACKAGE OCCUPIES FIELD 1 LOCATIONS 0 TO 1100.
FIELD 1 IS NEEDED ONLY IF THE CONSOLE PACKAGE IS BEING USED.

2.3 PRELIMINARY DIAGNOSTIC PROGRAMS
//////////

THIS PROGRAM ASSUMES THAT THE HOST PDP IS FUNCTIONING CORRECTLY.

3. LOADING PROCEDURE
////////////////////////////////

THIS PROGRAM IS IN BINARY FORMAT. TO LOAD THIS PROGRAM INTO CORE,
FOLLOW THE INSTRUCTIONS PUBLISHED FOR THE PARTICULAR BINARY FORMAT
LOADER BEING USED.

4. STARTING PROCEDURE
////////////////////////////////

4.1 STARTING ADDRESS LOCATIONS
////////////////////////////////

THIS PROGRAM HAS 2 STARTING ADDRESS LOCATIONS:

200 * DIAGNOSTIC PROGRAM *

STARTING THIS PROGRAM AT PROGRAM LOCATION 200, AND SUCCESSFULLY SUPPLYING
THE NECESSARY TEST PARAMETERS REQUESTED, DIRECTS THIS PROGRAM TO BEGIN
TESTING THE RX01 SUBSYSTEM AS A SUBSYSTEM OR PARTIAL SUBSYSTEM.
THE ADDRESS THAT THE CONSOLE PACKAGE WILL START THE DIAGNOSTIC
AT IS LOCATION 200.

201 * RESTART OF DIAGNOSTIC PROGRAM *

STARTING THIS PROGRAM AT PROGRAM LOCATION 201 DIRECTS THE PROGRAM TO
CONTINUE TESTING THE RX01 SUBSYSTEM AS A SUBSYSTEM OR PARTIAL SUBSYSTEM
USING THE PARAMETERS SUPPLIED AT STARTING ADDRESS 200.

4.2 OPERATOR ACTION
////////////////////////////////

CONFIGURE THE ACCUMULATOR SWITCHES TO REPRESENT THE OCTAL PROGRAM STARTING
ADDRESS LOCATION DESIRED, PRESS " LOAD ADDRESS ", PRESS " CLEAR ", AND
FINALLY PRESS " CONT " INUE. IF THE CONSOLE PACKAGE IS ACTIVE

THEN ISSUEING THE R DIRXA COMMAND WILL LOAD THE PROGRAM FROM
A DISKETTE.

↳ loads and starts

4.3 PROGRAM ACTION

////////////////////

THE PROGRAM PRINTS THE TITLE OF THE PROGRAM AND THE PRESENT
MAINDEC REVISION. IF THE PROGRAM STARTING ADDRESS IS 200 THE
PROGRAM WILL REQUEST TEST PARAMETER INFORMATION TO BE SUPPLIED BY
BY THE OPERATOR VIA THE PDP SWITCHES.

* 201 *

THE PROGRAM HAS RECONFIRMED PREVIOUS TEST PARAMETER SELECTIONS BY PRINTING
APPROPRIATE MESSAGES ON THE TELEPRINTER, AND HAS RESUMED TESTING.

4.4 TEST PARAMETER SELECTIONS

////////////////////

AFTER STARTING THIS PROGRAM AT STARTING ADDRESS 200, THE PROGRAM WILL PRINT
"REMOVE DIAGNOSTIC DISKETTE" (REMOVE AND REPLACE WITH SCRATCH DISKETTE)
" SELECT PARAMETERS (INCLUDING DEVICE CODE) " ~~AND HALT OR PRINT~~
THE SWITCH REGISTER QUESTION IF THE CONSOLE PACKAGE IS ACTIVE.

*blank in
each drive*

THE OPERATOR WILL THEN CONFIGURE CONSOLE SWITCHES 0-2 TO REPRESENT
DISKETTE UNIT/DRIVE SELECTION, CONSOLE SWITCHES 3, 4, AND 5 TO REPRESENT
RX8 INTERFACE IOT DEVICE CODE, AND FINALLY CONSOLE SWITCHES 7 - 11 TO
REPRESENT THE [STARTING] TEST , AND THEN PRESS " CONT " INUE.(SEE RESTRICTIONS)

*↑
"return"*

*Put in: 0000 for
all tests of both drives*

C O N S O L E
S W I T C H E S

0	1	2	3	4	5	6	7	8	9	10	11
U	U	U/C	X	X	X	U	T	T	T	T	T

X=DEVICE CODE (0 DEFAULTS TO 75)
T=TESTS (0 FOR ALL TESTS)
C=RX01 MICROCONTROL CABLED TO RX8 INTERFACE
U=UNIT/DRIVES SELECTED FOR TEST
WHERE: UNIT DRIVE(S)
0000 = A 0,1
1000 = B 0,1
2000 = A 0
3000 = B 0
4000 = A 1
5000 = B 1
6000 = TEST CONTROL AND INTERFACE ONLY (NO DRIVES)
7000 = TEST CONTROL ONLY (NO CONTROL OR DRIVES)
0040 = TEST UNITS A AND B (ALL DRIVES)
2040 = DRIVES 0 BOTH UNITS
4040 = DRIVES 1 BOTH UNITS

THE PROGRAM WILL PRINT A CONFIRMATION MESSAGE, THEN BEGIN TESTING THE RX01 SUBSYSTEM AS A SUBSYSTEM OR PARTIAL SUBSYSTEM WITH THE FIRST TEST CONFIGURED WITHIN CONSOLE SWITCHES 7 THRU 11.

TO TEST THE RX01 DISKETTE SUBSYSTEM(RXA) AS A COMPLETE SUBSYSTEM, CONFIGURE CONSOLE SWITCHES 0, 1, AND 2 TO REPRESENT ZERO. THE PROGRAM WILL ASSUME ALL DRIVE(S) ARE READY (POWER APPLIED / DOOR CLOSED).

NOTE, HOWEVER, THAT TESTING THE RX01 DISKETTE SUBSYSTEM AS A PARTIAL SUBSYSTEM IS MORE THAN MERELY CONFIGURING CONSOLE SWITCHES 0, 1, AND 2 TO REPRESENT VALUES OTHER THAN ZERO.

IF ONLY THE CONTROL / INTERFACE PARTIAL SUBSYSTEM (NO DRIVES) IS THE THE UNIT UNDER TEST (UUT), THEN THE DISKETTE DRIVE(S) CABLE (IF ANY) MUST BE PHYSICALLY REMOVED FROM THE RX01 CONTROLLER, OR, THE DRIVES MUST BE RENDERED " NOT READY " (DOOR OPEN), THEN THE CONSOLE SWITCHES 0, AND 1 EACH MUST BE CONFIGURED TO REPRESENT A " 1 ".

IF ONLY THE INTERFACE PARTIAL SUBSYSTEM (NO DRIVES AND CONTROL) IS THE UUT THEN THE BC05-L CABLE FROM THE INTERFACE TO THE CONTROL MUST BE PHYSICALLY

REMOVED, ONLY THEN MAY CONSOLE SWITCHES 0, 1, AND 2 BE CONFIGURED TO REPRESENT A " 1 ".

4.4.1 RESTRICTIONS
////////////////////

A SELECT GROUP OF TESTS ARE DESIGNED TO CONFIRM THE RESULTS OF PREVIOUS TESTS. THE FOLLOWING IS AN OVERVIEW OF THOSE TESTS WHICH REQUIRE A PREVIOUS TESTS TO HAVE EXECUTED CORRECTLY.

...FOR THIS TEST
TO FUNCTION
CORRECTLY...

...THIS TEST MUST
HAVE PREVIOUSLY
EXECUTED CORRECTLY...

T14, T15 / T13

TEST 14, AND TEST 15 EMPTY THE SECTOR BUFFER IN 12-BIT MODE COMPARING THE CONTENTS TO THE EXPECTED DATA PATTERN FILLED BY TEST 13.

T17, T20 / T16

TEST 17, AND TEST 20 EMPTY THE SECTOR BUFFER IN 8-BIT MODE COMPARING THE CONTENTS TO THE EXPECTED DATA PATTERN FILLED WITHIN TEST 16.

T24, T25, T26 / T22

THE SECTOR BUFFER SHOULD REMAIN UNCHANGED AND CONTAIN THE DATA PATTERN FILLED WITHIN TEST 22 AFTER THE RX01 CONTROL ATTEMPTS TO SEEK SECTOR 0.

T27 / T22

THE CONTENTS OF THE SECTOR BUFFER SHOULD REMAIN UNCHANGED AND CONTAIN THE DATA PATTERN FILLED WITHIN TEST 22 AFTER THE RX01 CONTROLLER ATTEMPTS TO SEEK AN ILLEGAL TRACK GREATER THAN 114.

T32 / T31

TEST 32 ISSUES AN INITIALIZE WHICH PERFORMS AN " IMPLIED READ " OF TRACK 1 / SECTOR 1. THE PROGRAM COMPARES THE DATA FROM THE SECTOR BUFFER WHICH SHOULD BE EQUIVALENT TO THE DATA WRITTEN WITHIN TEST 31 (FIRST WRITE EVER).

TEST 33 ISSUES THE FIRST READ EVER OF THE TRACK / SECTOR WRITTEN WITHIN
 TEST 31 AND COMPARES THE DATA EXPECTING THE DATA PATTERN OF TEST 31.
 5. OPERATING PROCEDURE
 //////////////////////////////////

THE OPERATOR MAY CONTROL THE DYNAMIC ACTION OF THE PROGRAM BY APPROPRI-
 ATELY CONFIGURING THE ACCUMULATOR SWITCHES AS DESCRIBED BELOW.
 IF THE CONSOLE PACKAGE IS ACTIVE THEN THE SETTING OF THE
 SWITCH REGISTER UNDER DYNAMIC CONDITIONS IS POSSIBLE BY
 TYPING CONTROL G THIS WILL PRINT THE SWITCH REGISTER
 QUESTION AND ALLOW CHANGES TO IT THEN CONTINUE RUNNING.

0 = 1 - INHIBIT ERROR HALT
 1 = 1 - LOCK SCOPE LOOP ON ERROR
 2 = 1 - LOCK SCOPE LOOP ON TEST (OK OR NOT)
 3 = 1 - HALT AT THE END OF A TEST PASS
 4 = 1 - INHIBIT ERROR PRINTOUTS
 5 = 1 - (LONG) DATA COMPARISON ERROR PRINTOUT
 6 = 1 - INHIBIT THE ISSUING OF [INIT] AT ERROR
 11 = 1 - DISABLE THE RINGING OF THE BELL AT AN ERROR

6. ERROR DETECTION

6.1 PROGRAM DEFINITIONS

THIS PROGRAM HAS DEFINED THE FOLLOWING AS ERRORS:

6.1.1 WRITE ERROR //////////////////////////////////

A WRITE ERROR IS A READ ERROR IF THE DATA BEING READ IS OF UNKNOWN
 QUALITY (THE DATA BEING READ IS BEING READ FOR THE FIRST TIME AFTER ITS
 WRITING) .

6.1.2 READ (CRC) ERROR - (TRANSFER REGISTER STATUS BIT 11 = 1) //////////////////////////////////

A READ ERROR IS A READ ERROR WHERE THE QUALITY OF THE DATA BEING READ
 IS KNOWN (THE DATA BEING READ HAD BEEN READ SUCCESSFULLY SOME TIME PRE-
 VIOUSLY).

6.1.3 CRC AND DATA ERROR

////////////////////////////////////

6.1.4 NO CRC BUT DATA ERROR

////////////////////////////////////

6.1.5 CRC BUT NO DATA ERROR

////////////////////////////////////

THESE DATA ERRORS ARE DETECTED WHEN THE PROGRAM IS VERIFYING THE DATA THAT " SHOULD HAVE BEEN READ " WITH THE DATA THAT " ACTUALLY WAS READ " BY COMPARING THE " BAD " COLUMN TO THE " GOOD " COLUMN.

WORD# GOOD BAD

1 (TRACK IDENTIFICATION BITS 5-11)
2 (SECTOR IDENTIFICATION BITS 8-11)

WORDS 3 THRU 62 (IF 12-BIT MODE), OR
BYTES 3 THRU 126 (IF 8-BIT MODE) CONTAIN
THE OPERATORS PPP SELECTION.

63 (OR BYTE 127) - THE SUM OF ALL WORDS 1 THRU 62
OR BYTES 1 THRU 127.

64 OR (BYTE 128) - THE NEGATIVE OF 2 TIMES
THE VALUE OF WORD # 63 OR BYTE # 127.

6.1.5.1 SUMCHECK ERROR

////////////////////////////////////

THE PROGRAM DETECTS A " SUM-CHECK " ERROR BY SUMMING ALL THE ACTUAL (BAD) DATA COLUMN AND COMPARING THAT SUM TO 0.

THE REASON FOR THE FIRST 2 WORDS/BYTES CONTAINING TRACK / SECTOR IDENTIFICATION CODES IS TO DETECT ADDRESSING ERRORS.

THE REASON FOR THE LAST 2 WORDS/BYTES CONTAINING CHECKSUM INFORMATION IS TO DISTINGUISH BETWEEN WHAT MIGHT RESEMBLE AN ADDRESSING ERROR (IF THE PROGRAM DETECTED AN ERROR WHEN COMPARING THE FIRST 2 WORDS/BYTES) AND A CRC ERROR.

6.1.6 SEEK ERROR - (NO ALLOCATED TRANSFER REGISTER STATUS BIT)
////////////////

A SEEK ERROR HAS BEEN DEFINED AS " NOT A CRC ", AND " NOT A PARITY " ERROR.

6.1.7 PARITY ERROR - (TRANSFER REGISTER STATUS BIT 10 = 1)
////////////////

A PARITY ERROR IS AN ERROR WHICH RESULTS FROM AN INCORRECT TRANSFER OF THE COMMAND WORD FROM THE RX8 INTERFACE TO THE RX01 MICROPROCESSOR CONTROL.

6.2 DEFINITIVE ERROR CODES
////////////////

THE RX01 MICROCONTROLLER HAS DEFINED ERROR CODES AND MEANINGS WHICH ARE AVAILABLE TO THE PROGRAM BY ISSUING COMMAND #7 TO " READ THE B-CODE ".

A DEFINITIVE ERROR CODE REPRESENTS [WHERE] WITHIN A MICRO-FUNCTION THE ERROR WAS DETECTED.

THE FOLLOWING ARE THE DEFINITIVE ERROR CODES AND MEANINGS:

0	-	NO ERROR
10	-	DRIVE 0 FAILED TO SEE HOME FROM INITIALIZE
20	-	DRIVE 1 FAILED TO SEE HOME FROM INITIALIZAE
30	-	HOME FOUND WHEN STEPPING OUT 10 TRACKS FROM INIT
40	-	TRIED TO ACCESS A TRACK GREATER THAN 77(DECIMAL)
50	-	HOME WAS FOUND BEFORE DESIRED TRACK
60	-	SELF DIAGNOSTIC ERROR
70	-	DESIRED SECTOR NOT FOUND AFTER SAMPLING 52 HEADERS
100	-	WRITE PROTECT ERROR
110	-	MORE THAN 40US AND NO SEP CLOCK DETECTED
120	-	A PREAMBLE COULD NOT BE FOUND
130	-	PREAMBLE FOUND BUT NO ID MARK FOUND IN TIME
140	-	CRC ERROR ON SUPPUSIDLY GOOD HEADER
150	-	GOOD HEADER(NO CRC ERROR) BUT TRACK COMPARE ERROR
160	-	IDAM NOT FOUND IN TOME
170	-	DATA AM NOT FOUND IN TIME
200	-	DATA CRC ERROR
210	-	ALL PARITY ERRORS

6.3 UNEXPECTED OR MISSING ERROR CONDITIONS
////////////////////////////////////

6.3.1 MISSING DD MARK
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE PROGRAM EXPECTED A DELETED DATA MARK BUT NONE OCCURED.

6.3.2 UNEXPECTED DD MARK
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE PROGRAM HAD NOT EXPECTED A DELETED DATA MARK BUT ONE OCCURED.

6.3.3 MISSING ERROR FLAG
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE CONTENTS OF THE TRANSFER REGISTER AT DONE TIME ARE NOT 0, AND THE ERROR FLAG IS CLEARED.

6.3.4 UNEXPECTED RX01 IRQ
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE PROGRAM HAS NOT YET ENABLED THE RX0 INTERRUPT ENABLE FLIP-FLOP BUT AN INTERRUPT OCCURED.

6.3.5 DEVICE TEST HUNG
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE PROGRAM EXPECTS BUT FAILED TO RECIEVE A PROGRAM INTERRUPT REQUEST FROM THE RX01 SUBSYSTEM WITHIN AN ALLOTTED PERIOD OF TIME (APPROXIMATELY 4 SECONDS).

6.4 UNKNOWN IRQ
////////////////////////////////////

THIS ERROR MAY OCCUR WHEN THE PROGRAM HAS FAILED TO IDENTIFY THE DEVICE ISSUING A PROGRAM INTERRUPT REQUEST.

7. ERROR REPORTING

////////////////////

ALL ERRORS DETECTED WILL BE REPORTED IF AC SW 3 = 0. THE PROGRAM HAS TWO ERROR REPORTING SCHEMES.

7.1.1 EXAMPLES IF INTERFACE / CONTROL RELATED ERRORS

THE FOLLOWING INFORMATION IS PRINTED FOR ALL INTERFACE / CONTROL RELATED ERRORS.

ERR	FAT	FAST	-	EAC	GOOD	PASS
ERR						
FAT						
FAST						
-						
EAC						
GOOD						
PASS						

- PPROGRAM ADDRESS OF THE ERROR
- FIRST ADDRESS OF THE TEST IN ERROR
- FIRST ADDRESS OF THE SUBTEST WITHIN THE TEST
- SEE ASSEMBLY LISTING FOR MAP
- ERROR AC (ACTUAL) RESULT OF TEST
- EXPECTED RESULT OF TEST
- PASS # AT ERROR

7.1.2 EXAMPLES OF DRIVE / DATA RELATED ERRORS

THE FOLLOWING INFORMATION IS PRINTED FOR ALL DRIVE / DATA RELATED ERRORS.

CMND	XDR	CODE	RSTA	START	TARGET	PASS
CMND						
XDR						
CODE						
RSTA						
START						
TARGET						
PASS						

- COMMAND TO THE RX01 MICROCONTROLLER
- CONTENTS OF THE TRANSFER REGISTER AT ERROR/DONE
- DEFINITIVE ERROR CODE (VIA COMMAND #7)
- STATUS (VIA COMMAND #5)
- STARTING TRACK/SECTOR ACTUATOR POSITION
- TARGET TRACK/SECTOR ACTUATOR POSITION
- PASS # AT ERROR (TO 16777215 DECIMAL)

7.1.2.1 SEEK ERROR

CMND	XDR	CODE	RSTA	START	TARGET	PASS
0014	0100	0120	0300	[HOME]	1 1	
INIT	0	0120	0200	[HOME]	[HOME]	

A SEEK ERROR OCCURED WHILE TRYING TO " WRITE DELETED DATA " (CMND #14) ONTO TRACK 1 SECTOR 1.

COMD	- WRITE DELETED DATA
XDR	- DELETED DATA MARK
CODE	- A PREAMBLE COULD NOT BE FOUND
RSTA	- DRIVE READY + DELETED DATA
START	- HOME POSITION
TARGET	- TRACK 1, SECTOR 1

THEN THE PROGRAM ISSUED AN INITIALIZE AT AN ATTEMPT TO RECOVER FROM
THE " SEEK " ERROR.

CMND - MEANS IOT 67X7 (INIT) WAS ISSUED
XDR - MUST BE SEEK (NOT CRC OR PARITY)
CODE - A PREAMBLE COULD NOT BE FOUND
RSTA - DRIVE READY
START - HOME POSITION
TARGET - HOME POSITION

7.1.2.2 WRITE-CRC AND DATA ERROR

CMND	XDR	CODE	RSTA	START	TARGET	PASS
0026	0001	3200	0201	100,30	100,1	
WRITE-CRC AND DATA ERROR						
WORD	GOOD	BAD				
4	5435	5473				
5	6617	5437				
6	6303	4606				
SUMCHECK IS 1253						
TOTAL BAD=60						

WHILE READING SECTOR 7 OF TRACK 100 THE PROGRAM DETECTED A CRC ERROR.

THE PROGRAM EXPANDS THE STANDARD ERROR FORMAT TO INCLUDE DATA COM-
PARISON INFORMATION IF THE TEST IS A DATA COMPARISON TEST.

IF AC SWITCH 2 =1 THEN A [LONG] DATA COMPARISON PRINTOUT WOULD
HAVE OCCURRED OF [ALL] THE WORDS/BYTES IN ERROR.

THE WORD " WRITE " WITHIN THE EXPANSION MEANS THAT THE DATA OF
SECTOR 7 HAD NEVER BEEN READ BEFORE, THEREFORE THE PROGRAM ASSUMED IT
WAS WRITTEN INCORRECTLY.

7.1.2.3 READ-CRC AND DATA ERROR

CMND	XDR	CODE	RSTA	START	TARGET	PASS
0026	0001	0200	0201	100,30	100,1	
READ-CRC AND DATA ERROR						
WORD	GOOD	BAD				
4	5435	5477				
5	6617	5437				
6	6303	5406				
SUM-CHECK IS 1257						
TOTAL BAD=60						

WHILE SEEKING SECTOR 1 OF TRACK 100 THE PROGRAM DETECTED A CRC ERROR.
THE PROGRAM EXPANDS THE STANDARD ERROR FORMAT TO INCLUDE DATA COMPARISON
INFORMATION IF THE TEST IS A DATA COMPARISON TEST.

IF AC SWITCH 2 = 1 THEN A [LONG] DATA COMPARISON ERROR PRINTOUT WOULD
HAVE OCCURRED OF [ALL] THE WORDS/BYTES IN ERROR.

THE WORD " READ " WITHIN THE EXPANSION MEANS THAT THE DATA OF
SECTOR 7 HAD BEEN READ SOME TIME PREVIOUSLY, THEREFORE THE PROGRAM
ASSUMES THAT THE DATA WAS WRITTEN CORRECTLY BUT READ INCORRECTLY.

7.1.2.4 WRITE-CRC BUT NO DATA ERROR

CMND	XDR	CODE	RSTA	START	TARGET	PASS
0020	0001	0200	0201	100,30	100,1	

WRITE - CRC BUT NO DATA ERROR

A CRC ERROR WAS DETECTED AFTER READING SECTOR 1 OF TRACK 100.

THE WORD " WRITE " WITHIN THE DATA EXPANSION MEANS THAT THE DATA
OF SECTOR 1 TRACK 100 HAD NEVER BEEN READ BEFORE THEREFORE THE PROGRAM
ASSUMED IT WAS WRITTEN INCORRECTLY, AND BECAUSE THE PROGRAM DID NOT DETECT
A DATA COMPARISON ERROR, IT ASSUMED THAT THE 2 CRC CHARACTERS WERE WRITTEN
INCORRECTLY.

7.2 ERROR RECOVERY ////////////////////

THE PROGRAM WILL ATTEMPT TO RETRY ALL ERRORING FUNCTIONS 10 TIMES.
A RECOVERABLE ERROR (SOFT) IS ONE WHICH DISAPPEARS WITHIN 10 PROGRAM
RETRYS. AN UNRECOVERABLE ERROR (HARD) IS ONE WHICH REMAINS AFTER 10
PROGRAM RETRYS.

THE PROGRAM WILL ISSUE IOT 67X7 [INIT] FOR ALL ERRORS BUT
DEFINITIVE ERROR CODES 140, 200, AND 210.

140 - CRC ERROR ON SUPPOSIDELY GOOD HEADER
200 - DATA CRC ERROR
210 - PARITY ERROR

THESE ERRORS ARE (NOT) SEEK TYPE ACTUATOR RELATED ERRORS.
IF A HARD WRITE ERROR IS DETECTED THE PROGRAM ABORTS FURTHER TESTING
THIS PASS OF THAT TRACK/SECTOR BUT CONTINUES TESTING TRACKS, ON THE REMAINING
A SYMPATHETIC HARD READ ERROR MAY OCCUR (PATTERN DEPENDENT) IF THE
HARD WRITE ERROR HAD OCCURRED WITHIN A TEST WHICH WOULD EVENTUALLY READ
THAT SECTOR AND A SYMPATHETIC HARD DATA COMPARISON ERROR MAY OCCUR IF THAT
TEST WAS TO VERIFY THE DATA TO A KNOWN PATTERN.

7.3 FATAL ERROR RECOVERY ////////////////////////////////////

IF THE PROGRAM DETECTS ANY OF THE SUCCEEDING FATAL ERROR CONDITIONS,
THE PROGRAMS RECOVERY WILL BE THAT OF EXITING THE PRESENT TEST.

- (A) HARD PARITY ERROR
- (B) A SELECTED DRIVE BECOMING NOT READY
- (C) NO EXPECTED RX01 INTERRUPT REQUEST
- (D) MISSING ERROR FLAG
- (E) LOG OVERFLOW
- (F) DEVICE TEST HUNG.

8. MANUAL INTERVENTION ////////////////////////////////////

8.1 FIELD ENGINEERING TROUBLE SHOOTING AIDS ////////////////////////////////////

THE FIELD ENGINEER, BY ALTERING THE CONTENTS OF SPECIFIC PROGRAM
MAINTENANCE LOCATIONS, IS ABLE TO DIRECT THE PROGRAM TO PERFORM TESTING
UPON A PARTICULAR AREA [WINDOW] OF THE DISKETTE INSTEAD OF THE ENTIRE
SURFACE. THESE PROGRAM LOCATIONS ARE LABELED "OD" "ID", "FIRST, AND "LAST".

"OD" (OUTSIDE DIAMETER), PROGRAM LOCATION 30, IS THE INITIAL OUTER-
MOST TRACK THE PROGRAM WILL ACCESS.

"ID" (INSIDE DIAMETER), PROGRAM LOCATION 31, IS THE FINAL INNERMOST
INNERMOST TRACK THE PROGRAM WILL ACCESS.

"FIRST", PROGRAM LOCATION 32, IS THE FIRST SECTOR TO BE ACCESSED
OF A TRACK.

"LAST", PROGRAM LOCATION 33, IS THE LAST SECTOR TO BE ACCESSED OF A TRACK.

THE STANDARD ASSEMBLED CONTENTS OF THESE FIELD ENGINEERING MAINTENANCE LOCATIONS ARE:

*30

OD,	52	/INITIAL TRACK TO TEST
ID,	53	/FINAL TRACK TO TEST
FIRST,	1	/FIRST SECTOR OF A TRACK
LAST,	32	/LAST SECTOR OF A TRACK

THESE ARE THE ONLY FIELD ENGINEERING MAINTENANCE PROGRAM LOCATIONS DESIGNED TO BE EXTERNALLY ALTERED.

THE PROGRAM WILL PRINT A MAINTENANCE VERIFICATION MESSAGE IF THE CONTENTS OF THE MAINTENANCE LOCATIONS ARE NOT THE ASSEMBLED STANDARDS.

THE OCTAL CONTENTS OF THESE MAINTENANCE LOCATIONS MUST BE WITHIN THE RESTRICTED LIMITS WHICH ARE:

0 <= OD <= ID
0 <= ID <= 114
1 <= FIRST <= LAST
1 <= LAST <= 32

THE PROGRAM VERIFIES THE CONTENTS OF EACH MAINTENANCE LOCATION, THE PROGRAM WILL SET INTO THE MAINTENANCE LOCATION THE STANDARD VALUE IF THE DESIRED CONTENTS WERE NOT WITHIN THE REQUIRED SPECIFIED LIMITS.

NOTE, THAT TRACK 0 IS NOT INCLUDED WITHIN THE STANDARD [WINDOW] OF TESTABLE TRACKS. THE REASON FOR THIS IS NOT TO INADVERTENTLY DESTROY THE FORMATTED CONTENTS OF TRACK 0.

TO EXPAND THE WINDOW OF TESTABLE TRACKS INCLUDING TRACK 0, THE CONTENTS OF PROGRAM LOCATIONS " OD " MUST BE ZERO.

IN SUMMARY, IF THE CONTENTS OF PROGRAM LOCATIONS OD, ID, FIRST, AND LAST WERE 30, 30, 1, 1, RESPECTIVELY, THE PROGRAM WOULD PERFORM SELECTED TESTING ONLY UPON TRACK 30, SECTOR 1.

9. DIAGNOSTIC TEST STRATEGY

////////////////////////////////////

THE PDP-8 DIAGNOSTIC PACKAGE, WHICH CONTAINS THE RX8/RX01 DIAGNOSTIC PROGRAM, MAYDEC-08-DIRXA-, AND THE RX8/RX01 DATA RELIABILITY/EXERCISER PROGRAM, MAYDEC-08-DIRXB-, IS DESIGNED TO COMPLETELY TEST THE RX8/RX01 SUBSYSTEM WHEN EACH PROGRAM IS RUN TO ITS NATURAL COMPLETION COMMENCING WITH THE RX8/RX01 DIAGNOSTIC PROGRAM.

THE RX8 DIAGNOSTIC PROGRAM CONTAINS EFFICIENT MAINTENANCE SCOPING LOOPS, WHILE THE RX8/RX01 DATA RELIABILITY/EXERCISER PROGRAM CONTAINS DISKETTE COMPATABILITY VERIFICATION.

10. CONSOLE PACKAGE
//////////

TABLE OF CONTENTS

10.1.	ABSTRACT
10.2.	REQUIREMENTS
10.3.	RESTRICTIONS
10.4.	STANDARD OPERATING PROCEDURE
10.4.1	INITIALIZATION
10.4.2	CONTROL CHARACTERS
10.4.3	
10.4.4	SWITCH REGISTER MESSAGE
10.4.5	END OF PASS MESSAGE
10.5.	ERRORS
10.5.1	ERROR HALTS
10.6.	SWITCH REGISTER SETTINGS
10.6.1	OPERATING SWITCHES
10.6.2	ERROR SWITCHES
10.7.	LOCATION CHANGES
10.8.	PROGRAM DESCRIPTION
10.9.	DIALOGUE FOR CONSOLE PACKAGE
10.10.	LISTING

10.1. ABSTRACT

THE CONSOLE PACKAGE HAS BEEN ADDED TO THIS DIAGNOSTIC TO ALLOW THE PROGRAM TO RUN WITH NO HARDWARE SWITCH REGISTER AND TO HAVE COMMUNICATIONS WITH THE DIAGNOSTIC VIA A TERMINAL.
THE DIAGNOSTIC CAN BE RUN IN TWO MODES WITH THE CONSOLE PACKAGE . 1) RUNNING WITH THE CONSOLE PACKAGE ACTIVE - THIS ALLOWS THE OPERATOR CONTROL OF THE DIAGNOSTIC THROUGH THE TERMINAL. THE DIAGNOSTIC WILL ASK FOR THE VALUE OF THE PSEUDO SWITCH REGISTER, BEFORE CONTINUING WITH EXECUTION OF THE DIAGNOSTIC ALL ERROR WILL BE PRINTED ON THE TERMINAL AND THE NUMBER OF PASSES WILL BE PRINTED. THERE WILL BE NO HALTS EXECUTED .
2) CONSOLE PACKAGE NOT ACTIVE-THIS WILL RESULT IN THE USE OF HALTS FOR ERROR, HALTS AT END OF PASS IF SELECTED, USE OF THE HARDWARE SWITCH REGISTER , NOT ASKING THE SWITCH QUESTION.
VT79 MUST ALWAYS BE RUN WITH ACTIVE CONSOLE PACKAGE.

10.2. REQUIREMENTS

10.3. RESTRICTIONS

-
- 1) RUNNING THE CONSOLE PACKAGE REQUIRES THAT THE PSEUDO SWITCH REGISTER BE USED.
 - 2) ONCE RUNNING THE CONSOLE PACKAGE NONACTIVE AND NOW DESIRE TO RUN IT ACTIVE, ONE MUST RELOAD THE DIAGNOSTIC AND INITILIZE FOR A ACTIVE CONSOLE PACKAGE.
 - 3.) THE CONSOLE PACKAGE IS LOCATED IN FIELD 1 THERE MUST BE 8K OF CORE TO RUN THE CONSOLE PACKAGE.

10.4. STANDARD OPERATION PROCEDURE

10.4.1 INITIALIZATION

FOR A ACTIVE CONSOLE PACKAGE

- 1.) LOAD ADDRESS 0021
- 2.) SET TO 0000 INDICATOR FOR USING THE PSEUDO SWITCH REGISTER
- 3.) LOAD ADDRESS 0022

- 4.) SET SR3=1 (400) INDICATOR FOR USING A ACTIVE CONSOLE PACKAGE
- 5.) LOAD STARTING ADDRESS OF PROGRAM AND BEGIN

FOR A NON ACTIVE CONSOLE PACKAGE

- 1.) LOAD ADDRESS 0021
- 2.) SET SR0=1 (4000) TO INDICATE A HARDWARE SWITCH REGISTER
- 3.) LOAD ADDRESS 0022
- 4.) SET TO 0000 TO INDICATE A DEACTIVE CONSOLE PACKAGE
- 5.) LOAD STARTING ADDRESS OF PROGRAM AND BEGIN

FOR VT78

--- ----

- 1) LOAD PROGRAM INTO CORE.
- 2) USING ODT SET LOC 21 TO 0000
LOC 22 TO 1400.
- 3) SAVE

10.4.2 CONTROL CHARACTERS

CONTROL CHARACTERS ARE USED TO GIVE THE OPERATOR THE ABILITY TO PERFORM THE FOLLOWING FUNCTIONS.
NOTE: THE PROGRAM WILL RESPOND TO THE CONTROL CHARACTER IN FIVE (5) SECONDS OR LESS.

CONTROL G	PRINT UPARROW G, DO A CARRIAGE RETURN AND LINE FEED, AND PRINT SR=XXXX, WAIT FOR INPUT FROM OPERATOR.
CONTROL C	RETURN TO MONITOR
CARRIAGE RETURN	RETURN TO PROGRAM, NO CHANGE TO PSEUDO SWITCH REGISTER.
LINE FEED	RESTART PROGRAM , NO CHANGE TO PSEUDO SWITCH REGISTER.
1 TO 4 OCTAL DIGITS AND CARRIAGE RETURN	CHANGE PSEUDO SWITCH REGISTER AND RETURN TO THE PROGRAM.
1 TO 4 OCTAL DIGITS AND A LINE FEED	CHANGE PSEUDO SWITCH REGISTER AND RESTART THE PROGRAM.

1 TO 4 OCTAL DIGITS
AND A CONTROL G

NO CHANGE TO PSEUDO SWITCH REGISTER
SAME RESPONSE AS A CONTROL G.

ALL OTHER INPUT
(ILLEGAL CHARACTERS
OR 5 DIGITS)

NO CHANGE TO PSEUDO SWITCH REGISTER.
ECHO THE CHARACTER, PRINT A QUESTION
MARK, DO A CARRIAGE RETURN LINE FEED,
PRINT SR=XXXX, WAIT FOR OPERATOR INPUT.

CONTROL S

THIS IS USED TO INHIBIT TRANSMISSION
OF DATA TO THE OPERATORS TERMINAL.
WHEN TYPED WHILE THE PROGRAM IS RUNNING
MESSAGE PRINTING IS INHIBITED, BUT THE
PROGRAM WILL CONTINUE TO RUN UNTIL A
MESSAGE IS PENDING. IF TYPED WHILE A
MESSAGE IS IN PROGRESS, THE PRINTOUT
WILL BE INTERRUPTED. A CONTROL Q OR
CONTROL C WILL RESUME PROGRAM ACTIVITY.
THIS IS A NONPRINTING CHARACTER.

CONTROL Q

THIS ENABLES DATA TRANSMISSION TO
OPERATORS TERMINAL IF A PREVIOUS
CONTROL S WAS IN EFFECT. THIS IS
A NONPRINTING CHARACTER.

10.4.4

SWITCH REGISTER MESSAGE

THIS MESSAGE IS USED TO SETUP THE PSEUDO SWITCH REGISTER BEFORE PROGRAM EXECUTION TAKES PLACE (OR IN RESPONSE TO A "G"). THE SWITCH REGISTER IS SETUP WHEN A CARRIAGE RETURN IS TYPED

SR=0000 4000

UNDER SCORING INDICATES OPERATOR RESPONSE

10.4.5

END OF PASS

A INDICATION WILL BE GIVEN WHEN THE DIAGNOSTIC HAS MADE A SUCCESSFUL PASS. THE PRINT OUT WILL INDICATE THE DIAGNOSTIC MATNDEC NUMBER THE WORD PASS AND A FOUR DIGIT PASS NUMBER. A PASS WILL BE A TIME PERIOD RATHER THAN A PROGRAM PASS OF THE DIAGNOSTIC. THE TIME PERIOD WILL BE IN THE RANGE OF ONE (1) TO FIVE (5) MINUTES. IF THE DIAGNOSTIC MAKES A PROGRAM PASS IN THE 1 TO 5 MINUTE RANGE THEN THE PASS COUNT WILL BE THE SAME AS THE NUMBER OF PROGRAM PASSES. IF THE PROGRAM MAKES A PROGRAM PASS IN LESS THEN ONE MINUTE THEN THE PASS COUNT WILL NOT BE THE SAME AS THE PASS COUNTER THE PASS COUNTER WILL REFLECT MORE THEN ON PROGRAM PASS. THE NUMBER OF PROGRAM PASSES REQUIRED FOR "A PASS MESSAGE CAN BE FOUND IN LOCATION CALLED "CNTVAL". IF HALT AT END OF PASS IS SET THEN THE PASS MESSAGE WILL BE PRINTED AND A WAITING STATEMENT WILL ALSO BE PRINTED. A CONTROL CHARACTER IS NEEDED TO CONTINUE FROM THIS MESSAGE. THE FORMAT OF THE END OF PASS MESSAGE IS

NAME PASS 0001 (- OR I OR C OR D)
- AN ERROR OCCURED DURING I,C,D
I INTERFACE TEST OK RX8
C RX8 AND RX01 TESTED OK
D RX8 AND RX01 AND DRIVE TESTED OK

10.5. ERRORS

UPON DETECTION OF A ERROR THE DIAGNOSTIC WILL DO ONE OF THE FOLLOWING OPERATIONS:

- 1.) PRINT THE ERROR MESSAGE FOR THE ERROR CONDITION
CHECK THE SWITCH REGISTER TO SEE IF THE PROGRAM SHOULD HALT
IF HALT ON ERROR IS SELECTED THEN THE PSR WILL BE PRINTED.
REFER TO THE LISTING AT THE LOCATION PRINTED IN THE ERROR
PC FOR THE CAUSE OF THE ERROR.

10.5.1 ERROR HALTS

CONSOLE PACKAGE DEACTIVE WILL CAUSE NO ERROR MESSAGE TO BE PRINTED, A HALT WILL REPLACE THE ERROR CALL IN THE CODE AND THE DIAGNOSTIC WILL THEN GO TO THAT HALT.
REFER TO THE LISTING FOR THE CAUSE OF THE ERROR, THE ERROR LOCATION WILL BE THE SAME IF THE CONSOLE PACKAGE WAS ACTIVE.

10.6. SWITCH REGISTER SETTINGS

THE FOLLOWING SWITCH REGISTER SETTINGS ARE USED BY THE CONSOLE PACKAGE, THESE SWITCH REGISTER SETTINGS ARE VALID WHEN USING THE HARDWARE SWITCH REGISTER AND THE PSEUDO SWITCH REGISTER.

10.6.1 OPERATING SWITCHES

THE CONSOLE PACKAGE USES THE LOCATIONS 20 21 22 FOR THE FOLLOWING PURPOSES.

LOCATION 20
PSEUDO SWITCH REGISTER

LOCATION 21
HARDWARE IDENTIFIER 1

LOCATION 22
HARDWARE IDENTIFIER 2

LOCATION 0022

BIT	OCTAL VALUE	FUNCTION WHEN 0	FUNCTION WHEN 1
---	-----	-----	-----
0	4000	NOT ON ACT 8A LINE	ON ACT 8A LINE
1	2000	NOT ON ACT 8E LINE	ON ACT 8E LINE
2	1000	NOT ON VT78	ON VT78
3	400	DEACTIVE CONSOLE PACKAGE	ACTIVE CONSOLE PACKAGE

6.2 ERROR RELATED SWITCHES

THESE ARE THE SWITCH REGISTER SETING THAT THE CONSOLE PACKAGE
WILL RECOGNISE.

BIT	OCTAL VALUE	FUNCTION WHEN 0	FUNCTION WHEN 1
---	-----	-----	-----
0	4000	STOP AFTER ERROR	CONT AFTER ERROR
1	2000	NO SCOPE LOOP ON ERROR	LOCK SCOPE LOOP ON ERROR
2	1000	NO SCOPE LOOP ON TEST	LOCK SCOPE LOOP ON TEST(UK OR NOT)
3	0400	CONTINUE AFTER END OF PASS	STOP AFTER END OF PASS
4	0200	PRINT ERROR MESSAGES	DO NOT PRINT ERRORS
5	0100	SHORT ERROR PRINTOUT	LONG DATA COMPARISON
6	0040	ISSUE[INIT]AT ERROR	DO NOT ISSUE [INIT] AT ERROR
11	0001	RING BELL ON ERROR	NO BELL RINGING

10.7. LOCATION CHANGES

THE FOLLOWING LOCATIONS CAN BE CHANGED TO MEET THE SPECIFIC
NEED FOR MODIFICATION OF THE DIAGNOSTIC.

CNTVAL IS THE LOCATION FOR THE VALUE OF THE
 NUMBER OF PROGRAM PASSES NEED TO
 PRINT THE END OF PASS MESSAGE.

FILLER IS THE LOCATION SET FOR THE NUMBER OF
 FILLER CHARACTERS AFTER A CRLF SET TO FOUR (4)

10.8. PROGRAM DESCRIPTION

PARAMETER SELECTIONS:

BITS 0-1

SELECT DRIVES: 0000 =SELECT DRIVES 0 AND 1(RXA)

SEE SECTION 4.4 TEST PARAMETER SELECTION

BITS 3-5

DEVICE CODE SELECTION

PUT ONE DIGITE DEVICE CODE IN HERE

THE PROGRAM ASSUMES THE FIRST DIGIT TO BE 7

EXAMPLE :IF DEVICE CODE IS 75 ENTER A 500

BITS 7-11

TEST SELECTION

PUT IN TEST YOU WISH TO RUN

A 00 TEST SELECTION IS ALL TESTS

10.9. DIALOGUE FOR CONSOLE PACKAGE

THE DIALOGUE FOR THIS DIAGNOSTIC IS:

R DTRXA (CR)

MAINDEC-08-DTRXA-D

/THE -D IS THE REVISION LEVEL

REMOVE DIAGNOSTIC DISKETTE

/REMOVE AND REPLACE

/WITH A SCRATCH DISKETTE

SELECT PARAMETERS(INCLUDING DEVICE CODES)

SR=0000

/DEVICE CODE SELECTED IS 75

/RXA DRIVES 0 AND 1

/AND ALL TESTS SELECTED

TEST PARAMETERS: 0000

/VERIFICATION FOR PARAMETERS

OD=0001 ID=0114 FIRST= 0001 LAST= 0032 /DISKETTE RANGE SELECTED

SR=0000 0400

/0400= SWITCH SETTING STOP ON ERROR AND

/AT END OF PASS

11. APT-8 INTERFACES

11.1 DESCRIPTION

TWO INTERFACES HAVE BEEN PROVIDED WHICH WILL ALLOW THIS DIAGNOSTIC TO RUN UNDER THE STANDARD APT-8 SYSTEM. THESE INTERFACES ARE:

1. TIMING INTERFACE
2. ERROR INTERFACE

EACH WILL BE EXPLAINED IN MORE DETAIL.

11.2 SETUP

IN ORDER TO RUN UNDER APT-8, ADDRESSES 20 AND 22 MUST BE ESTABLISHED PRIOR TO RUNNING THE PROGRAM UNDER APT-8 CONTROL. THE FOLLOWING INFORMATION MUST BE INDICATED:

1. DEVICE CODE OF RX01 CONTROLLER UNDER TEST.
2. RX01 CONTROLLER CABLED TO INTERFACE.
3. DRIVE OR DRIVES TO BE TESTED.
4. DIAGNOSTIC RUNNING UNDER THE APT-8 SYSTEM.
5. STARTING TEST NUMBER IF OTHER THAN THAT FOR ACCEPTANCE.

ADDRESS 20

ADDRESS 20 IS USED TO ESTABLISH ALL BUT ITEM 4. THE SET UP IS THE SAME FOR THAT OF THE STANDARD SWITCH REGISTER FUNCTION.

0 1 2 3 4 5 6 7 8 9 10 11
U U C X X X - T T T T T
(STARTING TEST)

- 0 = 0 - DISKETTE DRIVE(S) IS READY
- 1 = 0 - (POWER APPLIED / DOOR CLOSED)
- 2 = 0 - RX01 MICROCONTROL CABLED TO RX8 INTERFACE
- 3 = - FOR STANDARD DEVICE CODE
- 4 = - POSITION CONSOLE SWITCHES
- 5 = - 3-4-5 = 0
- 7 = - TO EXECUTE ALL TESTS
- 8 = - CONFIGURE CONSOLE SWITCHES
- 9 = - 7 THRU 11
- 10 = - EQUIVALENT TO ZERO
- 11 = - (THIS IS THE STARTING TEST)

NOTE: IF MORE THAN ONE DEVICE CODE IS AVAILABLE ON THE SYSTEM THE DIAGNOSTIC WILL HAVE TO RELOADED AND THE PROPER DEVICE CODE SELECTED.

ADDRESS 22

THIS ADDRESS IS USED TO INDICATE THAT THE PROGRAM IS RUNNING ON APT-8, THE NUMBER OF DRIVES TO BE DONE, AND IF SINGLE DRIVE TESTING.

BIT ZERO MUST BE A ONE (1) TO INDICATE THAT THE DIAGNOSTIC IS RUNNING ON APT-8.

CAUTION:

WHILE UNDER APT-8 CONTROL THE HARDWARE SWITCH REGISTER IS FUNCTIONAL. IT IS RECOMMENDED THAT THE SWITCH REGISTER BE SET TO ZERO PRIOR TO START-UP.

11.3 APT-8 INTERFACES.

11.3.1 TIMING

APT-8 IS NOTIFIED OF PROGRAM RUN BETWEEN .2 SECONDS AND 2.0 SECONDS. THIS WILL ALLOW THE DIAGNOSTIC TO RUN UNDER THE MUCH SLOWER MOS MEMORY WITHOUT CAUSING APT-8 TO GIVE A TIMEOUT ERROR.

11.3.2 ERRORS

ONLY THE ERROR PC IS REPORTED TO APT-8. THE TYPE OF ERROR CAN BE DETERMINED FROM THE CORRESPONDING ADDRESS IN THE PROGRAM LISTING. THERE IS A POSSIBILITY THAT A TIMEOUT ERROR MAY OCCUR. THIS IS CAUSED BY THE ERROR "HUNG DEVICE". THE PROGRAM WILL HAVE TO BE RERUN IN DUMP MODE IF THIS SHOULD HAPPEN.

11.4 LOADING PRECAUTIONS

THIS PROGRAM SHOULD BE LOADED IN SRCIPT MODE INDICATING TO APT-8 THAT CORE SUMCHECKS ARE TO BE IGNORED.

12. PROGRAM LISTING

```

1 /RX8 RX01 DIAGNOSTIC DIRXA-D
2 0001 FIELD 1
3 / CONSOL PACKAGE
4
5
6
7
8 /PROGRAM SHOULD CHECK FOR A CONTROL CHARACTER FROM THE CONSOL
9 /EVERY FIVE SECONDS OR LESS
10
11
12
13 /SETUP CNTVAL FOR A RANGE OF 1 TO 4(8/E) MINUTES FOR C8PASS TO PRINT PASS
14 /SETUP OF CNTVAL WILL BE FOUND IN C8PASS
15 /THIS VALUE SHOULD BE A POSITIVE NUMBER.
16
17 /CHANGE 1-7 APRIL ,1975
18
19 /VT78 MODIFICATIONS JUNE ,1977
20 /
21 / 1. NEW CONSOLE PACKAGE
22 / 2. CAPABILITY OF TESTING 2 FLOPPY UNITS(4 DRIVES)
23 / 3. MODIFIED INTERLEAVING
24
25 6661 PSKF= 6661
26 6662 PCLF= 6662
27 6663 PSKE= 6663
28 6664 PSTB= 6664
29 6665 PSIE= 6665
30 6004 GTF= 6004
31 7701 ACL= 7701
32 6007 CAF= 6007
33 7421 MQL= 7421
34 4461 HLT= HALT
35
36 /#6 MQA= 7501
37
38 /#6
39 /XLIST
40 /IFDEF CONSOL <
41 /XLIST
42 *24
43
44 4424 C8PASS= JMS I .
45 0024 0600 XC8PAS . /C8 PASS COMPLETION ROUTINE
46 4425 C8CKSW= JMS I .
47 0025 0673 XC8SW . /CHECK SW REG SETTING
48 4426 C8PRNT= JMS I .
49
50 0026 1000 XC8PMT . /C8 PRINT A BUFFER
51 4427 C8OCTA= JMS I .
52 0027 1035 XC8OCT . /CONVERT TO ASCII AND PRINT
53
54 /*20 /PSEUDO SWITCH REGISTER
55 /4000=DU NOT INHIBIT ERROR HALT
/2000=LOOP ON ERROR
/1000=LOOP ON TEST IN SR 4-11

```

```

56 /400=HALT AT END OF PASS
57
58 /*21 /HARDWARE INDICATORS
59 /4000=USE FRONT PANEL SWITCH REGISTER
60 /*22 /SYSTEM CONFIGURATION
61 /400=CONSOL PACKAGE SET ACTIVE
62 /*23 /RESERVED FOR FUTURE USE
63
64 /XLIST> C O N S O L E P A C K A G E
65 /
66 /
67 /IF ENTERED WITH C8CHAR=0000 THE SWITCH REGISTER
68 /MODIFICATION ROUTINE IS ENTERED AUTOMATICALLY.
69 /IF ENTERED WITH C8CHAR NOT EQUAL TO 0000, THE
70 /KEYBOARD INPUT DECODER IS ENTERED AND IT IS ASSUMED
71 /THAT THE AC CONTAINS THE ASCII CODE TO BE
72 /CHECKED FOR A VALID CONTROL CHARACTER.
73 /
74 /
75 /
76 /
77 /
78 7002 BSW=7002
79 /
80 0030 0000 C8TEMP, 0 /TEMPORARY WORK AREA
81 0031 0000 C8CHAR, 0
82 0032 6203 C8CDI, CIF CDF /USED TO CREATE CDI TO PRUGHAM FIELD
83 0033 6201 C8CDF, CDF /USED TO CREATE CDF TO CONSOLE FIELD
84 0034 0000 C8SWR, 0 /SWITCH REGISTER SAVE AREA
85 0035 0000 C8MODE, 0 /PRINT MODE SWITCH
86 0036 0000 C8CNTR, 0 /USED AS COUNTER
87 0037 7775 C8#3, -3 /CONSTANT
88 0040 7774 C8#4, -4 /CONSTANT
89 0041 7773 C8#5, -5 /CONSTANT
90 0042 7770 C8#10, -10 /CONSTANT
91 0043 7520 C8#260, -260 /CONSTANT
92 0044 0007 C8K7, 0007 /CONSTANT
93 0045 0240 C8K240, 0240 /CONSTANT
94 0046 0260 C8K260, 0260 /CONSTANT
95 0047 0275 C8K275, 0275 /CONSTANT
96 0050 0277 C8K277, 0277 /CONSTANT
97 0051 0322 C8K322, 0322 /CONSTANT
98 0052 0323 C8K323, 0323 /CONSTANT
99 /
100 /
101 *200
102
103 C8ENTR, 0
104 0201 1777 TAD I (21 /GET HCW1 FROM PROGRAM FIELD
105 0202 7710 SPA CLA /SKIP IF USING PSEUDO SWR
106 0203 7614 7614 /GET HARDWARE SWR AND SKIP
107 0204 1776 TAD I (20 /GET PSEUDO SWR
108 0205 3034 DCA C8SWR /SAVE SWITCH REGISTER
109 0206 1775 TAD I (INMODE /GET MESSAGE ACTIVE FLAG
110 0207 3035 DCA C8MODE /SAVE IT

```

```

111 0210 6211 CDF 10 /CHANGE DATA FIELD TO CONSOLE PACKAGE
112 0211 1200 TAD C8ENTR /GET RETURN ADDRESS
113 0212 3774* DCA C8RTN /SAVE FOR EXIT
114 0213 1031 TAD C8CHAR /GET CHARACTER SAVED IN MAIN PROGRAM
115 0214 7440 SZA /SKIP IF IT WAS ZERO
116 0215 5773* JMP C8CNTL /AC NOT ZERO, GO CHECK CTRL CHAR
117
118 /PRINT OUT SR=XXXX WHERE XXXX IS THE CURRENT CONTENTS
119 /OF THE SWITCH REGISTER BEING USED (EITHER PSEUDO OR HARDWARE)
120
121 0216 4772* C8PSW, JMS C8CRLF /DO A <CR> AND <LF>
122 0217 1052 TAD C8K323 /GET ASCII CODE FOR "S"
123 0220 4771* JMS C8TYP /PRINT "S"
124 0221 1051 TAD C8K322 /GET ASCII CODE FOR "R"
125 0222 4771* JMS C8TYP /PRINT "R"
126 0223 1047 TAD C8K275 /GET ASCII CODE FOR "="
127 0224 4771* JMS C8TYP /PRINT "="
128 0225 1040 TAD C8M4 /AC=-4
129 0226 3036 DCA C8CNTR /SET UP OCTAL DIGIT COUNTER
130 0227 1034 TAD C8SWR /GET SWITCH REGISTER
131 0230 7004 RAL /EXTRA ROTATE FOR LINK
132 0231 7004 C8LOPA, RAL
133 0232 7006 RTL /ROTATE OCTAL DIGITS FOR PRINTING
134 0233 3034 DCA C8SWR /SAVE ROTATED SWR
135 0234 1034 TAD C8SWR /GET ROTATED SWR
136 0235 0644 AND C8K7 /MASK OFF DIGIT TO PRINT
137 0236 1046 TAD C8K260 /ADD ASCII BASE CODE
138 0237 4771* JMS C8TYP /PRINT AN OCTAL DIGIT
139 0240 1034 TAD C8SWR /GET SWR
140 0241 2036 ISZ C8CNTR /INCREMENT LOOP COUNTER
141 0242 5231 JMP C8LOPA /GO PRINT NEXT DIGIT
142
143 /ACCEPT KEYBOARD INPUT OF OCTAL DIGITS, <CR>, <LF>
144 /CTRL/C OR CTRL/G. ALL OTHER CHARACTERS ARE INVALID
145 /AND WILL BE ECHOED, FOLLOWED BY A "?",
146 /A CARRIAGE RETURN, LINE FEED, AND A RESTART OF
147 /THE SR=XXXX ROUTINE
148
149
150 0243 7300 CLA CLL
151 0244 1041 TAD C8MS /AC=-5
152 0245 3036 DCA C8CNTR /SET UP TO ACCEPT 5 CHARACTERS
153 0246 3773* DCA C8BLD /CLEAR SWITCH REG. BUILD AREA
154 0247 3767* DCA C8FLG /CLEAR SWR CHANGE SWITCH
155 0250 1045 TAD C8K240 /GET ASCII CODE FOR SPACE
156 0251 4771* JMS C8TYP /SPACE OVER ONE POSITION
157 0252 4766* C8SRLP, JMS C8TTY /GO WAIT FOR KEYBOARD INPUT
158
159 0253 3030 DCA C8TEMP /SAVE INPUT CHARACTER
160 0254 1030 TAD C8TEMP /GET CHARACTER
161 0255 1365 TAD (=203
162 0256 7450 SNA /SKIP IF NOT CTRL/C
163 0257 5764* JMP C8CTLC /GO TO CTRL/C ROUTINE
164 0260 1040 TAD C8M4 /AC=-4
165 0261 7450 SNA /SKIP IF NOT CTRL/G
166 0262 5763* JMP C8CTLG /GO TO CTRL/G ROUTINE

```

```

166 0263 1037 TAD C8M3 /SUBTRACT 3
167 0264 7450 SNA /SKIP IF NOT LINE FEED
168 0265 5762* JMP C8EXT1 /GO TO LINE FEED EXIT
169 0266 1037 TAD C8M3 /SUBTRACT 3
170 0267 7650 SNA CLA /SKIP IF NOT CARRIAGE RETURN
171 0270 5761* JMP C8EXT2 /GO TO CARRIAGE RETURN EXIT
172 0271 1030 TAD C8TEMP /GET CHARACTER
173 0272 4771* JMS C8TYP /ECHO 11
174 0273 1030 TAD C8TEMP /GET CHARACTER
175 0274 1043 TAD C8M260
176 0275 7510 SPA /SKIP IF >= TO ASCII CODE FOR ZERO
177 0276 5316 JMP C8ERR /INVALID CHARACTER NOT OCTAL DIGIT
178 0277 1042 TAD C8M10
179 0300 7700 SNA CLA /SKIP IF <= ASCII CODE FOR SEVEN
180 0301 5316 JMP C8ERR /INVALID CHARACTER NOT OCTAL DIGIT
181 0302 7240 STA /AC=7777
182 0303 3767* DCA C8FLG /SET SWR CHANGE FLAG
183 0304 1030 TAD C8TEMP /GET CHARACTER
184 0305 0044 AND C8K7 /MASK TO 3 BITS
185 0306 3030 DCA C8TEMP /SAVE OCTAL DIGIT
186 0307 1770* TAD C8BLD /GET SWR BUILD AREA CONTENTS
187 0310 7106 CLL RTL
188 0311 7004 RAL /ROTATE TO BUILD SWR
189 0312 1030 TAD C8TEMP /ADD NEXT OCTAL DIGIT
190 0313 3770* DCA C8BLD /SAVE NEW SWR
191 0314 2036 ISZ C8CNTR /INCREMENT OCTAL DIGIT COUNTER
192 0315 5252 JMP C8SRLP /CONTINUE ACCEPTING OCTAL DIGITS
193
194 0316 7300 C8ERR, CLA CLL
195 0317 1050 TAD C8K277 /GET ASCII CODE FOR "?"
196 0320 4771* JMS C8TYP /PRINT "?"
197 0321 4772* JMS C8CRLF /DO A <CR> AND <LF>
198 0322 5216 JMP C8PSW /GO START OVER
199
200
201 0361 0541
202 0362 0531
203 0363 0457
204 0364 0465
205 0365 7575
206 0366 0514
207 0367 0403
208 0370 0400
209 0371 0502
210 0372 0523
211 0373 0420
212 0374 0402
213 0375 0351
214 0376 0020
215 0377 0021
216 0400 PAGE
217 0400 0000 /C8BLD, 0 /SWITCH REGISTER BUILD AREA
218 0401 0200 C8SRLP, RSTART /ADDRESS OF START OF PROGRAM
219 0402 0000 C8RTN, 0 /STORAGE FOR RETURN ADDRESS

```



```

220 0403 0000 C8FLG, 0 /SWP CHANGE SWITCH
221 0404 0000 C8SFLG, 0 /CTRL/S ACTIVE FLAG
222 0405 0177 C8K177, 0177 /CONSTANT
223 0406 0200 C8K200, 0200 /CONSTANT
224 0407 0077 C8K77, 0077 /CONSTANT
225 0410 7740 C8M40, =40 /CONSTANT
226 0411 0100 C8K100, 0100 /CONSTANT
227 0412 0215 C8K215, 0215 /CONSTANT
228 0413 0212 C8K212, 0212 /CONSTANT
229 0414 0303 C8K303, 0303 /CONSTANT
230 0415 0307 C8K307, 0307 /CONSTANT
231 0416 0336 C8K336, 0336 /CONSTANT
232 0417 7600 C87600, 7600 /CONSTANT
233 /
234 / CONTROL CHARACTER
235 /
236 / DECODE ROUTINE
237 /
238 0420 1377 C8CNTL, TAD (-203
239 0421 7450 SNA /SKIP IF NOT CTRL/C
240 0422 5265 JMP C8CTLC /CTRL/C TYPED EXIT TO MONITOR
241 0423 1376 TAD (-4
242 0424 7450 SNA /SKIP IF NOT CTRL/G
243 0425 5257 JMP C8CTLG /CTRL/G TYPED GO PRINT "G"
244 0426 1375 TAD (-12
245 0427 7450 SNA /SKIP IF NOT CTRL/Q
246 0430 5255 JMP C8CTLQ /CTRL/Q TYPED
247 0431 1374 TAD (-2
248 0432 7450 SNA /SKIP IF NOT CTRL/S
249 0433 5237 JMP C8CTLS /CTRL/S TYPED
250 0434 3035 OCA /SET MESSAGE ACTIVE FLAG
251 0435 2204 ISZ C8SFLG /TEST CTRL/S ACTIVE FLAG
252 0436 5275 JMP C8ECHO /GO ECHO CHARACTER AND RETURN TO PROGRAM
253 /
254 /CTRL/S HANDLER
255 /
256 0437 7240 C8CTLS, STA /AC=7777
257 0440 3204 DCA C8SFLG /SET CTRL/S ACTIVE FLAG
258 0441 1035 TAD C8MODE /GET MESSAGE ACTIVE FLAG
259 0442 7650 SNA CLA /SKIP IF CTRL/S TYPED WHILE MESSAGE ACTIVE
260 0443 5342 JMP C8PFLD=1 /RETURN TO PROGRAM
261 /
262 0444 7240 C8WAIT, STA /AC=7777
263 0445 3204 DCA C8SFLG /SET CTRL/S ACTIVE FLAG
264 0446 4314 JMS C8TTY /WAIT FOR KEYBOARD INPUT
265 0447 1377 TAD (-203
266 0450 7450 SNA /SKIP IF NOT CTRL/C
267 0451 5265 JMP C8CTLC /CTRL/C TYPED EXIT TO MONITOR
268 0452 1373 TAD (-16
269 0453 7640 SZA CLA /SKIP IF CTRL/G
270 0454 5244 JMP C8WAIT /NOT CTRL/C OR CTRL/Q CONTINUE WAITING
271 0455 3204 C8CTLQ, DCA C8SFLG /CLEAR CTRL/S ACTIVE FLAG
272 0456 5342 JMP C8PFLD=1 /RETURN TO MAIN PROGRAM
273 /
274 /CONTROL G HANDLER

```

```

275 /
276 0457 4323 C8CTLG, JMS C8CRLF /DO A <CR> AND <LF>
277 0460 1216 TAD C8K336 /GET ASCII CODE FOR UP ARROW
278 0461 4302 JMS C8TYP /PRINT UP ARROW
279 0462 1215 TAD C8K307 /GET ASCII CODE FOR "G"
280 0463 4302 JMS C8TYP /PRINT "G"
281 0464 5772 JMP C8PSW /GO TO "SR=XXXX" ROUTINE
282 /
283 /CONTROL C HANDLER
284 /
285 0465 3204 C8CTLC, DCA C8SFLG /CLEAR CTRL/S ACTIVE FLAG
286 0466 1216 TAD C8K336 /GET ASCII CODE FOR UP ARROW
287 0467 4302 JMS C8TYP /PRINT UP ARROW
288 0470 1214 TAD C8K303 /GET ASCII CODE FOR "C"
289 0471 4302 JMS C8TYP /PRINT "C"
290 0472 6203 CIF CDF /CHANGE TO IF AND DF ZERO
291 0473 6007 CAF
292 0474 5617 JMP I C87600 /RETURN TO MONITOR
293 /
294 0475 1031 C8ECHO, TAD C8CHAR /GET CHARACTER
295 0476 4302 JMS C8TYP /ECHO IT
296 0477 1050 TAD C8K277 /CODE FOR ?
297 0500 4302 JMS C8TYP /TYPE QUESTION MARK
298 0501 5342 JMP C8PFLD=1 /RETURN TO PROGRAM
299 /
300 /
301 /
302 /
303 /PRINT ONE CHARACTER
304 /
305 0502 0000 C8TYP, 0
306 0503 2204 ISZ C8SFLG /TEST CTRL/S ACTIVE FLAG
307 0504 7410 SKP /SKIP IF CTRL/S NOT ACTIVE
308 0505 5244 JMP C8WAIT /GO WAIT FOR CTRL/Q OR CTRL/C
309 0506 6046 TIS /TRANSMIT CHARACTER
310 0507 6041 TSF /TEST TTY FLAG
311 0510 5307 JMP *-1 /WAIT FOR TTY FLAG
312 0511 6042 TCF /CLEAR TTY FLAG
313 0512 7200 CUA /CLEAR AC DO NOT CLEAR LINK
314 /
315 0513 5702 JMP I C8TYP /RETURN
316 /
317 /WAIT FOR KEYBOARD INPUT THEN EXIT WITH ASCII CODE IN AC
318 /
319 0514 0000 C8TTY, 0
320 0515 6031 KSF /SKIP IF KEYBOARD FLAG SET
321 0516 5315 JMP *-1 /WAIT FOR KEYBOARD INPUT
322 0517 6036 KRB /READ KEYBOARD BUFFER CLEAR FLAG
323 0520 0205 AND C8K177 /MASK TO 7 BITS
324 0521 1206 TAD C8K200 /SET BIT 4
325 0522 5714 JMP I C8TTY /RETURN
326 /
327 /EXECUTE A CARRIAGE RETURN AND LINE FEED
328 /
329 0523 0000 C8CRLF, 0

```

```

330 0524 1212 TAD C8K215 /GET ASCII CODE FOR CARRIAGE RETURN
331 0525 4302 JMS C8TYP /GO EXECUTE THE CARRIAGE RETURN
332 0526 1213 TAD C8K212 /GET ASCII CODE FOR LINE FEED
333 0527 4302 JMS C8TYP /GO EXECUTE THE LINE FEED
334 0530 5723 JMP I C8CRLF /RETURN
335
336 /CONSOLE PACKAGE EXIT IF TERMINATED WITH LINE FEED
337
338 0531 4323 C8EXT1, JMS C8CRLF /DO A <CR> AND <LF>
339 0532 3031 DCA C8CHAR /CLEAR IT
340 0533 6203 CDF CIF 00 /MODIFIED CDI TO PROGRAM FIELD
341 0534 2203 ISZ C8FLG /TEST SWR CHANGE FLAG
342 0535 5601 JMP I C8STRT /RESTART PROGRAM WITHOUT CHANGE OF SWR
343 0536 1200 TAD C8BLD /GET NEW SWITCH REGISTER
344 0537 3771 DCA I (20 /SAVE IT IN PROGRAM FIELD
345 0540 5601 JMP I C8STRT /RESTART PROGRAM WITH NEW PSEUDO SWR
346
347 /EXIT FROM CONSOLE PACKAGE IF TERMINATED WITH CARRIAGE RETURN
348
349 0541 4323 C8EXT2, JMS C8CRLF /DO A <CR> AND <LF>
350 0542 3031 DCA C8CHAR /CLEAR IT
351 0543 6203 C8PFLD, CDF CIF 00 /MODIFIED CDI TO PROGRAM FIELD
352 0544 7300 CLA CLL /CLEAR AC AND LINK FOR RETURN
353 0545 2203 ISZ C8FLG /TEST SWR CHANGE FLAG
354 0546 5602 JMP I C8RTN /RETURN TO PROGRAM WITHOUT CHANGE OF SWR
355 0547 1200 TAD C8BLD /GET NEW SWITCH REGISTER
356 0550 3771 DCA I (20 /SAVE IT IN PROGRAM FIELD
357 0551 5602 JMP I C8RTN /RETURN TO PROGRAM
358
359 /
360 /
361 0571 0020
362 0572 0716
363 0573 7762
364 0574 7776
365 0575 7766
366 0576 7774
367 0577 7575
368 PAGE
369 0600 0600
370 /*****
371 /C8PASS
372 /THIS IS CALLED AT THE END OF EACH PROGRAM COMPLETIUN
373 /THE VALUE OF** CNTVAL** WILL BE DETERMINED BY THE TIME IT TAKES
374 /THE PROGRAM TO COMPLEATE THIS MANY C8PASS TO BE IN THE 1 TO 4 MINUTE
375 /RANGE
376 / C8PASS=JMS XC8PAS
377 /EX. C8PASS
378
379 / HLT /HALT IF NON CONSOL PACKAGE
380 / JMS START1 /CONTINUE RUNNING THIS PROGRAM
381 /RETURN TO LOCATION CALL PLUS ONE WITH THE AC=0 IF NON CONSOL PACKAGE AND HLT
382 /IF CONTINUE TO RUN THEN RETURN TO CALL PLUS2 AC=0
383
384 0600 0000 XC8PAS, 0
385 0601 6211 CDF 10 /CHANGE DATA FIELD TO CONSOLE PACKAGE

```

```

384 0602 4777 JMS C8CK22 /CHECK IF CONSOL IS ACTIVE
385 0603 5211 JMP DOPACK /IS CLASSIC
386 0604 4273 JMS XC8SW /CHECK SR SETTING
387 0605 0376 AND (400 /FOR HALT ON END OF C8PASS
388 0606 7640 SZA CLA /I= HALT 0 CONTINUE
389 0607 4461 HLT
390 0610 5240 JMP C8BY1 /CONTINUE ON RUNNING PROGRAM
391 0611 4242 DOPACK, JMS CKCOUT /CLASS CHECK C8PASS COUNT
392 0612 5240 JMP C8BY1 /C8PASS COUNT NOT DONE REDO PROGRAM
393 0613 2261 ISZ PASCNT /C8PASS COUNT DONE SET C8PASS COUNT
394
395 0614 4775 JMS C8CRLF /#2
396
397 0615 4774 JMS XC8PNT /C8PNT BUFFER
398 0616 0664 MESPAS /
399 0617 1261 TAD PASCNT /GET NUMBER
400 0620 4773 JMS XC8OCT /CONVERT IT TO ASCII
401 0621 6201 CDF 0
402 0622 1657 TAD I XMX /GET THE CHAR TO PRINT
403 0623 6211 CDF 10
404 0624 3242 DCA CKCOUT /STORE FOR PRINTING
405 0625 4774 JMS XC8PNT /PRINT IT
406 0626 0642 CKCOUT
407 0627 4775 JMS C8CRLF /DO A CARRIAGE RETURN
408 0630 4273 JMS XC8SW /CHECK A HALT AT END OF C8PASS
409 0631 0376 AND (400 /MASK BIT
410 0632 7650 SNA CLA /HALT =1 NO SKIP CONTINUE =0
411 0633 5240 JMP C8BY1 /NO HALT
412 0634 1200 TAD XC8PAS /GET RETURN ADDRESS
413 0635 3772 DCA C8ENTR /PLACE IT AT START OF CONSOLE PACKAGE
414 /ROUTINE = NEEDED FOR SETUP OF C8RTN
415 0636 6201 CDF 00
416 0637 5771 JMP C8ENTR+1 /GO TO CONSOLE PACKAGE TO PRINT PSR
417 0640 6203 C8BY1, CDF CIF 00
418 0641 5600 JMP I XC8PAS
419 0642 0000 CKCOUT, 0
420 0643 1262 TAD DOSET /CHECK IF SET UP NEEDED
421 0644 7640 SZA CLA /0=SET UP C8PASS COUNT VALUE
422 /1=C8PASS COUNT VALUE OK
423 /C8PASS COUNT VALUE OK
424 0645 5252 JMP NOSET /C8PASS COUNT VALUE OK
425 0646 1263 TAD CNTVAL /GET COUNT VALUE FOR THIS PROG
426 0647 7040 CMA /SET TO NEGATIVE
427 0650 3260 DCA DOCNT /STORE IN HERE
428 0651 2262 ISZ DOSET /INDICATE VALUE SET UP
429
430 /#2
431 0652 2260 NOSET, ISZ DOCNT /COUNT THE NUMBER OF PASSES
432
433 /#2
434 0653 5240 JMP C8BY1 /EXIT FOR ANOTHER PASS
435 0654 3262 DCA DOSET /SET TO C8PNT C8PASS
436 0655 2242 ISZ CKCOUT /BUMP RETURN FOR
437 0656 5642 JMP I CKCOUT /C8PASS C8TYPE OUT
438 0657 0522 XMX, MX /LOCATION OF CHAR TO PRINT AT PASS FROM FLD 0
439 0660 0000 DOCNT, 0
440 0661 0000 PASCNT, 0
441 0662 0000 DOSET, 0

```

439 0663 0000 CNTVAL, 0 ✓
440 0664 0411 MESPAS, TEXT "DIRXA-D PASS "
0665 2230
0666 0155
0667 0440
0670 2001
0671 2323
0672 4000

/C8CKSW
/ROUTINE THAT WILL CHECK WHERE TO READ THE
/C8 SWITCHES FROM IP, FROM PANEL OR PSEUDO C8SWIT REGISTER
/
/RETURN WITH AC = TO CONTENTS OF SWITCH REGISTER

452 0673 0000 XC8SW, 0
453 0674 7200 CLA /CLEAR AC
454 0675 6201 CDF 00
455 0676 1770 IAD I (21 /GET WD FOR INDICATOR
456 0677 6211 CDF 10
457 0700 7710 SPA CLA /CHECK IF FROM PANEL 4000
458 0701 7614 7614 /DO LAS AND SKIPGET FROM PANEL WITH LAS
459 0702 5304 JMP C8GET1 /GET CONTENTS OF LOC 20 FLD 00
460 0703 5307 JMP C8EXTB /EXIT COMMON
461 0704 6201 C8GET1, CDF 00
462 0705 1767 IAD I (20 /PSEUDO SW
463 0706 6211 CDF 10
464 0707 5673 C8EXTB, JMP I XC8SW /EXIT WITH STATUS BIT IN AC,
465
466
467 0767 0020
468 0770 0021
469 0771 0201
470 0772 0200
471 0773 1035
472 0774 1000
473 0775 0523
474 0776 0400
475 0777 1060
PAGE
1000

479 /C8PRNT
480
481
482
483 /THIS ROUTINE WILL TYPE THE CONTENTS OF THE C8 PRINT BUFFER, THE LOCATION
484 /OF THE BUFFER WILL BE IN THE ADDR8 FOLLOWING THE CALL, C8 PRINTING OF THE BUFFER
485 /WILL STOP WHEN A 00 CHAR IS DETECTED, CHARACTERS ARE PACKED 2 PER WORD.
486

487 / C8PRNT= JMS XC8PNT

490 /EX. C8PRNT /C8PRNT THE CONTENTS OF THE FOLLOWING BUFFER
491 / MESS77 /LOCATION OF C8PRNT BUFFER
492 /C8PRNT WILL USE THE LOCATION FOLLOWING THE CALL AS THE POINTER FOR THE
493 /C8PRNT ROUTINE, RETURN TO CALL PLUS TWO WITH AC= 0

498 1000 0000 XC8PNT, 0
499 1001 7300 CLA CLL
500 1002 1600 TAD I XC8PNT /GET C8PRNT BUFFERS STARTING LOCATION
501 1003 3233 DCA PISTOR /STORE IN PISTOR
502 1004 2200 ISZ XC8PNT /BUMP RETURN
503 1005 1633 C8D01, TAD I PISTOR /GET DATA WORD
504 1006 0377 AND (7700 /MASK FOR LEFT BYTE
505 1007 7450 SWA /CHECK IF 00 TERMINATE
506 1010 5600 JMP I XC8PNT /EXIT
507 1011 7500 SWA /IS AC MINUS
508 1012 7020 CML /MAKE CHAR A 300 AFTER ROTATE
509 1013 7001 IAC /MAKE CHAR A 200 AFTER ROTATE
510 1014 7012 RTR
511 1015 7012 RTR
512 1016 7012 RTR /PUT CHAR IN BITS 4-11 MAKE IT 8 BIT ASCII
513 1017 4770 JMS C8TYP /C8PRNT IT ON CONSOLE
514 1020 1633 TAD I PISTOR /GET DATA WORD
515 1021 0375 AND (0077 /MASK FOR RIGHT BYTE
516 1022 7450 SWA /CHECK IF 00 TERMINATOR
517 1023 5600 JMP I XC8PNT //EXIT
518 1024 1374 TAD (3740 /ADD FUDGE FACTOR TO DETERMINE IF 200
519 1025 7500 SWA /OR 300 IS TO BE ADD TO CHAR
520 1026 1373 TAD (100 /ADD 100
521 1027 1372 TAD (240 /ADD 200
522 1030 4770 JMS C8TYP /C8TYPE ONLY BITS 4-11
523 1031 2233 ISZ PISTOR /BUMP POINTER FOR NEXT WORD
524 1032 5205 JMP C8D01 /DO AGAIN
525 1033 0000 PISTOR, 0 /STOR FOR C8PRNT BUFFER
526 1034 0000 STOPNT, 0 /0000 C8PRNT 7777=DO NOT C8PRNT

534 /C8OCTA
535 /OCTAL TO ASCII CONVERSION
536 /THIS ROUTINE WILL TAKE THE OCTAL NUMBER IN THE AC AND CONVERT IT TO ASCII
537 /THE RESULT WILL BE PRINTED ON THE CONSOL DISPLAY
538 / C8OCTA= JMS XC8OCT
539 /
540 /EX. C8OCTA /AC CONTAINS NUMBER TO BE CHANGE
541 /

```

542
543 1035 0000 XC8OCT, 0
544 1036 7106 CLL RTL
545 1037 7006 RTL /POSITION THE FIRST CHAR FOR PRINTING
546 1040 3256 DCA C8TMP1 /SAVE CORRECT POSITIONED WORD HERE
547 1041 1371 TAD (-4
548 1042 3257 DCA C8CKP /STORE COUNTER IN HERE
549 1043 1256 C8DO4, TAD C8TMP1 /GET FIRST NUMBER
550 1044 0370 AND (0007 /MASK
551 1045 1367 TAD (260 /ADD THE PRINT CONSTANT
552 1046 4776 JMS C8TYP /TYPE THE NUMBER
553 1047 1256 TAD C8TMP1 /
554 1050 7006 RTL
555 1051 7004 RAL /PUT NEXT NUMBER IN POSITION
556 1052 3256 DCA C8TMP1 /STORE IT
557 1053 2257 ISZ C8CKP /DONE YET WITH FOUR NUMBERS
558 1054 5243 JMP C8DO4 /NOT YET DO MORE
559 1055 5635 JMP I XC8OCT /DONE WITH FOUR
560 1056 0000 CRTMP1, 0
561 1057 0000 C8CKP, 0
562
563 /*****
564
565 /*****
566 /CHECK LOCATION 22 FIELD 0
567
568
569 1060 0000 C8CK22, 0
570 1061 7200 CLA
571 1062 6201 CDF 00
572 1063 1766 TAD I (22 /GET LOC 22 FIELD 0
573 1064 6211 CDF 10
574 1065 0365 AND (400
575 1066 7650 SNA CLA
576 1067 2260 ISZ C8CK22
577 1070 5660 JMP I C8CK22 /EXIT
578
579
580
581
582
583 /
584 1165 0400
585 1166 0022
586 1167 0260
587 1170 0007
588 1171 7774
589
589 1172 0240
590 1173 0100
591 1174 3740
592 1175 0077
593 1176 0502
594 1177 7700
FIELD 0

```

```

0000 00000000 00000000 00001111 11111111 11111111 11100000 00000000 00000000
0100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11100000 00000000 00000000 00000000 01111111 11111111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0500 11111111 11111111 11111111 11111111 11111111 11000000 00000000 01111111

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 00000000 00000000 00000000 00000000 00000000 00000001 11111111

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 10000000
1100 00000000 00000000 00000000 00000000 00000000 00000000 00000111 11111111

1200
1300

1400
1500

1600
1700

2000
2100

2200
2300

2400
2500

2600
2700

3000
3100

3200
3300

3400
3500

3600
3700

```

4000
4100

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

```

595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649

```

```

/ RX8/RX01 DIAGNOSTIC * MAINDEC-08-DIRXA-D
/
/STARTING ADDRESS 200 = (AC) = STARTUP SWITCHES
/
/STARTING ADDRESS 201 = RESTART (PARAMETERS ALREADY SELECTED AT START 200)
/
/STARTING ADDRESS 202 = CHANGE DEVICE CODES.
/
/START-UP AC SWITCH DEFINITIONS (WHEN THE STARTING ADDRESS IS 200)
/
AC 0 1 2 3 4 5 6 7 8 9 10 11
   U U U/C X X X U T T T T T
/
   X=DEVICE CODE (0 DEFAULTS TO 75)
   T=TESTS (0 FOR ALL TESTS)
   C=RX01 MICROCONTROL CABLED TO RX8 INTERFACE
   U=UNIT/DRIVES SELECTED FOR TEST
   WHERE:      UNIT      DRIVE(S)
/
/           0000 = A      0,1
/           1000 = B      0,1
/           2000 = A      0
/           3000 = B      0
/           4000 = A      1
/           5000 = B      1
/           6000 = TEST CONTROL AND INTERFACE ONLY (NO DRIVES)
/           7000 = TEST INTERFACE ONLY (NO CONTROL OR DRIVES)
/           VT78 0040 = TEST UNITS A & B (ALL DRIVES)
/           4040 = DRIVES 1 BOTH UNITS
/           2040 = DRIVES 0 BOTH UNITS
/
/ THE PRETEST IS NOT SWITCH SELECTABLE - THIS PRETEST IS ALWAYS
/ EXECUTED PRIOR TO THE FIRST TTTT SELECTION
/
/ PRETEST = VERIFICATION OF INIT [KEY]
/
/ (PRETEST) IB - INIT PART I [KEY] / FLAG DETECTION PART I
/
/ INTERFACE / CONTROL TESTS
/
/ NOTE: * THE KEYBOARD IS * ALIVE * DURING TESTS 6, 7, 10, AND 11, AND 12
/ IF A CHARACTER IS STRUCK AN ERROR MAY OCCUR
/
TTTT
/
/ 0 I - FLAG DETECTION PART II / " C " LINES PART I
/ 1 IB - DIRECTION OF IOT XDR PART I / IOT DECODING PART I
/ " C " LINES PART II
/ 2 IB - FLAG DETECTION PART II / " C " LINES PART III
/ 3 IB - IOT DEVICE CODE VERIFICATION
/ 4 IB - DIRECTION OF IOT XDR PART II / " C " LINES PART IV
/ 5 I - IOT DECODING PART II

```

```

650 / * 6 IB/I - INTERRUPT TESTING PART I / IOT DECODING PART III
651 / * 7 IB - INTERRUPT TESTING PART II
652 / * 10 IB - INTERRUPT TESTING PART III
653 / * 11 I - INTERRUPT TESTING PART IV
654 / * 12 IB/I - INIT PART II [PROGRAMMED] / INTERRUPT TEST PART V
655 /RX01 CONTROL TESTS
656 /
657 / 13 C - FILL BUFFER 12-BIT MODE
658 / 14 C - EMPTY BUFFER 12-BIT MODE
659 / 15 C - VERIFICATION OF PREVIOUS TEST
660 / 16 C - FILL BUFFER 8-BIT MODE
661 / 17 C - EMPTY BUFFER 8-BIT MODE
662 / 20 C - VERIFICATION OF PREVIOUS TEST
663 / 21 C - FILL BUFFER 8-BIT MODE (ALL 0'S)
664 / 22 C - FILL BUFFER 8-BIT MODE (ALL 1'S)
665 /
666 /DISKETTE DRIVE TESTS
667 /
668 / 23 D - STATUS BIT "DRIVE READY"
669 / 24 D - B-CODE VERIFICATION (70) PART I
670 / 25 D - B-CODE VERIFICATION (70) PART II
671 / 26 D - B-CODE VERIFICATION (70) PART III
672 / 27 D - B-CODE VERIFICATION (40) PART IV
673 / 30 D - SEEK AND CRC VERIFICATION
674 / 31 D - WRITE TEST
675 / 32 D - INIT PART III [PROGRAMMED] IMPLIED READ TRACK 1 SECTOR 1
676 / 33 D - READ TEST
677 / 34 D - WRITE-READ-PROGRAM VERIFY 12 BIT MODE
678 / 35 D - WRITE-READ-PROGRAM VERIFY 8 BIT MODE
679 / 36 D - WRITE-READ-PROGRAM VERIFY 12 BIT MODE WITH DELETED DATA
680 / 37 D - WRITE-READ-PROGRAM VERIFY 8 BIT MODE WITH DELETED DATA
681 /
682 / I - MEANS RX8 INTERFACE TEST
683 / B - MEANS RX01 MAY BE CABLED TO RX8
684 / C - MEANS AN RX01 MUST BE CABLED TO THE RX8
685 / D - MEANS A DRIVE MUST BE READY
686 /
687 /OPERATIONAL AC SWITCH DEFINITIONS
688 /
689 / AC 0 - (1) CONTINUE ON DETECTION OF ERROR
690 / AC 1 - (1) LOCK SCOPE LOOP ON ERROR
691 / AC 2 - (1) LOCK SCOPE LOOP ON TEST
692 / AC 3 - (1) HALT AT END OF PASS
693 / AC 4 - (1) DON'T PRINT AN ERROR MESSAGE
694 / AC 5 - (1) LONG DATA COMPARISON ERROR PRINTOUT
695 / AC 6 - (1) DISABLE THE ISSUING OF [INIT]
696 / AC 7 - (1) HALT AT END OF A TEST
697 /
698 / AC 8 -
699 / AC 9 -
700 / AC 10 -
701 / AC 11 - (1) DISABLE RINGING OF BELL AT ERROR
702 / THE FOLLOWING MAP IS A SUMMARY OF ALL ERRORS.
703 /
704 /

```

ERA	TEST	BLANK	EAC	GOOD	COMMENT:
-----	------	-------	-----	------	----------

```

705 /
706 /
707 / 1700 0412 EOPRE / PRETEST - - - UNEXPECTED TRANSFER REQUEST FLAG
708 / 1701 0415 E2PRE / - - - UNEXPECTED ERROR FLAG
709 / 1702 0425 E3PRE / - - - MISSING DONE FLAG
710 / 1703 0445 E1PRE / STATUS FROM TR STATUS DEL DATA 4, 204 STATUS NOT = INIT DONE, OR DRIVE READY + INIT DONE
711 /
712 /
713 /
714 / 1704 0450 E4PRE / - - - UNEXPECTED DONE FLAG
715 /
716 /
717 / 1205 0607 E0 / T0 - X 0 IOT 67X1 DIDN'T CLEAR AC
718 / 1706 0613 E1 / - - - UNEXPECTED TRANSFER REQUEST FLAG
719 / 1707 0617 E2 / - - - UNEXPECTED ERROR FLAG
720 / 1210 0623 E3 / - - - UNEXPECTED DONE FLAG
721 /
722 /
723 / 1211 0647 E11 / T1 - X 0 IOT 67X1 FAILED TO CLEAR AC
724 / E10 / - X 200 TR NOT = 200 (DRIVE READY)
725 /
726 /
727 / 1212 0660 E20 / T2 - - - MISSING DONE FLAG
728 / E21 / - - - MISSING TR FLAG
729 / 1213 0663 E22 / - - - MISSING ERROR FLAG
730 / 1214 0666 E23 / - - - MISSING DONE FLAG
731 / 1215 0671 E24 / - - - MISSING TR FLAG
732 / 1216 0674 E25 / - - - MISSING ERROR FLAG
733 / 1217 0704 E26 / - X 7777 IOT 67X6 CLEARED AC
734 / 1220 0712 E27 / - X 7777 IOT 67X3 CLEARED AC
735 / 1221 0720 E28 / - X 7777 IOT 67X4 CLEARED AC
736 /
737 /
738 /
739 / 1222 0744 E30 / T3 - X 67X5 (EAC) = ILLEGAL DEVICE CODE
740 /
741 /
742 /
743 / 1223 1021 E42 / T4 7776 X 200 IOT 67X1 CLEARED AC
744 /
745 /
746 / 1224 1044 E40 / TO 376
747 / 375 DATA TO TR NOT =
748 / 373 DATA FROM TR
749 / 1225 1060 E41 / 7677 367 OR:
750 / 357 DATA FROM TR NOT =
751 / 337 DATA FROM TR PREVIOUSLY
752 / 7677
753 / 1226 1102 E56 / T5 - X 0 (TR) NOT = 0
754 / 1227 1105 E50 / - - - MISSING DONE FLAG
755 / 1230 1111 E53 / - - - UNEXPECTED DONE FLAG
756 / 1231 1114 E51 / - - - MISSING TR FLAG
757 / 1232 1120 E54 / - - - UNEXPECTED TR FLAG
758 / 1233 1123 E52 / - - - MISSING ERROR FLAG
759 / 1234 1127 E55 / - - - UNEXPECTED ERROR FLAG

```

```

760 /
761 /
762 /
763 1235 1142 E60 / T6 - - - UNEXPECTED RX01 IRQ
764 1236 1154 E61 / - - - MISSING DONE FLAG
765 1237 1157 E62 / - - - MISSING TR FLAG
766 1240 1162 E63 / - - - MISSING ERROR FLAG
767 /
768 /
769 /
770 1241 1212 E70 / T7 - - - MISSING RX01 IRQ
771 /
772 /
773 /
774 1242 1226 E100 / T10 - - - UNEXPECTED RX01 IRQD
775 /
776 /
777 /
778 1243 1252 E110 / T11 - - - UNEXPECTED RX01 IRQ
779 /
780 /
781 /
782 1244 1276 E124 / T12 - - - UNEXPECTED RX01 IRQ
783 1245 1311 E120 / - - - UNEXPECTED DONE FLAG
784 1246 1315 E121 / - - - UNEXPECTED TR FLAG
785 1247 1321 E122 / - - - UNEXPECTED ERROR FLAG
786 1250 1326 E123 / - X 0 (TR) NOT = 0
787 1251 1412 EA120 / ALT 12 - - - UNEXPECTED TR FLAG
788 1252 1416 EA121 / - - - UNEXPECTED DONE FLAG
789 1253 1440 EA122 / ACTUAL ACTUAL 4 OR
790 / STATUS MINUS 204 STATUS NOT = 4, OR 204
791 / DEL DAT
792 1254 1451 EA123 / X 0 THE B-CODE NOT = 0
793 /THE " XRSTB " SUBROUTINE WHICH READS THE B-CODE STATUS BY ISSUING
794 /COMMAND # 7 IS ENTERED FROM TESTS; *** ALT12, (T24, T25, T26), AND T27
795 /
796 /THE CONTENTS OF " BLANK " = THE CONTENTS OF GOOD FROM TEST; ALT12
797 /
798 1255 2405 E7000 / 4/204 - - UNEXPECTED TR FLAG
799 1256 2416 E7001 / 4/204 STATUS - UNEXPECTED ERROR FLAG
800 1257 2432 E7002 / 4/204 X 7000 SHIFT REGISTER NOT SHIFTING
801 /
802 /
803 /
804 /THE " XRST " SUBROUTINE TO READ THE STATUS REGISTER BY ISSUING COMMAND # 5
805 /IS ENTERED FROM TESTS; *** ALT12, AND T23
806 /
807 1260 2446 E7003 / 4/204 - - UNEXPECTED TR FLAG
808 1261 2457 E7004 / 4/204 STATUS - UNEXPECTED ERROR FLAG
809 1262 2473 E7006 / 4/204 STATUS 5000 SHIFT REGISTER NOT SHIFTING
810 1263 2511 E7005 / 4/204 STATUS 0/200
811 /
812 /
813 /
814 /SUBROUTINE " FBEB " TO FILL AND EMPTY THE BUFFER IS ENTERED FROM TESTS;

```

```

815 / T13, T16 (FILL THE BUFFER) / T14, T17 (EMPTY THE BUFFER)
816 /
817 1264 1513 E130 / COMMAND - - UNEXPECTED ERROR FLAG
818 1265 1526 E131 / COMMAND ACTUAL EXPECT # OF TR FLAGS NOT OK
819 / (EAC) = # OF FLAGS
820 / NEG. # MEANS NOT ENOUGH
821 / >0 MEANS TO MANY
822 /
823 /
824 /
825 /ERROR # 140 MAY OCCUR WITHIN TESTS T14, T17, T15, AND T20
826 /
827 /THE CONTENTS OF " BLANK " IS EQUIVALENT TO THE WORD/BYTE COUNT AT THE ERROR
828 /
829 /THE CONTENTS OF THE " EAC " IS EQUIVALENT TO THE ACTUAL DATA FROM THE SECTOR
830 /BUFFER (8 OR 12-BIT MODE)
831 /
832 /THE CONTENTS OF " GOOD " IS EQUIVALENT TO THE EXPECTED CONTENTS OF THE
833 /SECTOR BUFFER
834 /
835 1266 1661 E140 / * # ACTUAL EXPECT DATA COMPARISON ERROR
836 / " FB128BYTES " IS A SUBROUTINE WHICH FILLS THE SECTOR BUFFER WITH 128 BYTES
837 /OF DATA (ALL 1'S OR ALL 0'S) AND IS ENTERED FROM TESTS T21, AND T22
838 /
839 1267 4557 E210 / * - - - UNEXPECTED ERROR FLAG
840 /
841 /
842 /SUBROUTINE " TX " EMPTIES THE SECTOR BUFFER AND COMPARES THE DATA TO AN
843 /EXPECTED PATTERN
844 /
845 /THIS SUBROUTINE IS ENTERED FROM TESTS; *** T21, T22, (T24, T25, T26), AND T27
846 /
847 1270 2320 E211 / # ACAUAL EXPECT DATA COMPARISON ERROR
848 1271 2332 E212 / - - - UNEXPECTED ERROR FLAG
849 /
850 /
851 /
852 /ERRORS E240, E245, E241, AND E242 MAY OCCUR WITHIN TESTS; T24, T25, T26
853 /
854 /
855 /
856 1272 2030 E240 / * CMND X 2 # OF TR FLAGS NOT OK
857 1273 2033 E245 / CMND - - MISSING ERROR FLAG
858 1274 2070 E241 / ACTUAL X T24/200,300
859 / T25/200
860 / T26/300
861 1275 2101 E242 / - X 70 B-CODE NOT = 70
862 /
863 /
864 /
865 1276 2135 E270 / T27 115 # OF 2 # OF TR FLAGS NOT OK
866 / XFERS
867 1277 2140 E271 / - - - MISSING ERROR FLAG
868 1300 2151 E272 / - X 40 B-CODE NOT = 40
869 / *0

```

```

870 0000 0303          303          / C IS THE REVISION
871          0001          *1
872 0001 5402          JMP I IPI
873
874          /PROGRAM LOCATION 2 CONTAINS THE INTERRUPT RETURN ADDRESS
875          /
876          /PROGRAM LOCATION 2 IS MODIFIED WITHIN CERTAIN TESTS
877          /
878 0002 5417          IPI,   PI
879          /
880          /
881          /AUTO INDEX REGISTER DEFINITION.
882          /
883          0010          *10
884 0010 0000          A10,   0
885 0011 0000          A11,   0
886 0012 0000          A12,   0
887 0013 0000          A13,   0          /TEST Q
888          0020          *20
889          /
890          /THE FOLLOWING PROGRAM LOCATIONS (20, 21, AND 22, 23) ARE RESERVED FOR ACT8/A
891          /
892 0020 0000          0000          /SET FOR DRIVES 0 AND 1 DEVICE CODE 75
893 0021 4000          4000          /0000=PSEUDO SWITCH REGISTER IF ON ACTIVE CONSOLE
894          /0000=VT78
895          /4000=USE HARDWARE SWITCH REGISTER
896
897
898 0022 0000          0000          /0000=NOT ACTIVE CONSOLE PACKAGE
899          /0400= ACTIVE CONSOLE PACKAGE
900          /1400= VT78
901          0024          *24
902
903          /*****
904          /THE FOLLOWING CALLS ARE USED FOR THE CONSOLE PACKAGE
905          /
906          /
907          4424          CHECKC8= JMS I .
908 0024 4516          XC8ECK          /USED TO CHECK IF THE CONSOLE IS ACTIVE
909          /
910          4425          XC8ENTR= JMS I .
911 0025 0200          C8ENTP          /
912          4426          C8PASS= JMS I .
913 0026 0600          XC8PASS          /END OF PASS FOR CONSOLE
914          /
915          0200          RSTART=200          /RESTART ADDRESS FOR CONSOLE PACKAGE
916          /*****
917          /
918          /THE FOLLOWING PROGRAM LOCATIONS "OD", "ID", "FIRST", AND "LAST" MAY BE
919          /CHANGED BY THE OPERATOR MANUALLY HOWEVER FOLLOWING THESE RESTRICTIONS.
920          /
921          /          1. THE CONTENTS OF " OD " (MIN VAL 0) MUST BE <= THE
922          /          CONTENTS OF " ID " (MAX VAL 114).
923          /
924          /          2. THE CONTENTS OF "FIRST" (MIN VAL 1) MUST BE <= THE

```

```

925          /          CONTENTS OF "LAST" (MAX VAL 32)
926          /
927          /THE PROGRAM INITIALLY SETS THESE VALUES AT PROGRAM LOAD TIME
928          /
929          / (OD) = 52, AND (ID) = 53
930          /
931          /BECAUSE TRACK 53 IS THE TRACK AT WHICH THE RX01 MICROCONTROLLER WILL
932          /DECREASE THE WRITE CURRENT IN HALF
933          /
934 0027 0001          OD,    1          /OUTSIDE DIAMETER (MIN VALUE 0)
935 0030 0114          ID,    114         /INSIDE DIAMETER (MAX VALUE 114)
936 0031 0001          FIRST, 1          /FIRST SECTOR TO ACCESS (MIN VAL 1)
937 0032 0032          LAST,  32         /LAST SECTOR TO ACCESS (MAX VAL 32)
938          /
939          /PDP-8/E AUGMENTED INSTRUCTIONS
940          /
941          4433          BSW=JMS I .
942 0033 2600          XBSW          /BYTE SWAP
943          6001          ION=6001
944          6002          IOF=6002
945          4572          LAS=CKSWIT
946          7501          MQA=7501          / "OR" (MQ) WITH (AC)
947          7421          MQL=7421          /MQ=AC (THEN CLEAR AC)
948          /
949          /DISKETTE IOT SUBROUTINES
950          /
951          4434          TY8OCT=JMS I .
952 0034 5200          XTY8OCT          /TYPE EIGHT OCTAL DIGITS.
953          4435          SEL=JMS I .
954 0035 6200          XSEL
955          4436          LCD=JMS I .
956 0036 6203          XLCD
957          4437          LCDA=JMS I .
958 0037 6210          XLCDA
959          4440          LCDB=JMS I .
960 0040 6235          XLCDB
961          4441          XDRIN=JMS I .
962 0041 6400          XXDRIN
963          4442          XDROUT=JMS I .
964 0042 6403          XXDROUT
965          4443          STR=JMS I .
966 0043 6407          XSTR
967          4444          SER=JMS I .
968 0044 6414          XSER
969          4445          SDN=JMS I .
970 0045 6421          XSDN
971          4446          INTR=JMS I .
972 0046 6426          XINTR
973          4447          INIT=JMS I .
974 0047 6434          XINIT
975          4450          INITB=JMS I .
976 0050 6450          XINITB
977          4451          CKUNIT=JMS I .
978 0051 2335          XCKUNT
979          /OPERATING SYSTEM SUBROUTINES

```



```

980 /
981 4452 ERROR=JMS I .
982 0052 2625 XERROR
983 5453 EXIT=JMP I .
984 0053 0452 MORETESTS /EXIT FROM A TEST (IF RX8 ONLY)
985 4454 DONE=JMS I .
986 0054 4243 XDONE / FORM: "DONE; NO; YES"
987 4455 GETAPATTERN=JMS I .
988 0055 3035 XGETAPATTERN
989 4456 GETASECTOR=JMS I .
990 0056 4607 XGETASECIOP
991 4457 GETATRACK=JMS I .
992 0057 4261 XGETATRACK /GET A TRACK FOR IOT LCD-B (TRACK #)
993 4460 GETUNIT=JMS I .
994 0060 4200 XGETUNIT /SELECT A DISKETTE DRIVE
995 4461 HLT=HALT
996 4461 HALT=JMS I .
997 0061 4121 XHALT
998 4462 INITSECTORS=JMS I .
999 0062 4600 XINITSECTORS
1000 4463 INITTRACKS=JMS I .
1001 0063 4250 XINITTRACKS
1002 4464 LOCKUP=JMS I .
1003 0064 2545 XLOCKUP
1004 4453 NOTEST=EXIT
1005 4465 OK=JMS I .
1006 0065 6123 XOK /SKIP IF NOT ON APT.
1007 4466 TICK=JMS I .
1008 0066 4133 XTICK
1009 4467 AERROR=JMS I .
1010 0067 4346 XAERRO
1011 4470 APT8=JMS I .
1012 0070 1545 XAPT8
1013 4471 WAIT=JMS I .
1014 0071 1722 XWAIT
1015 4472 CHEK22=JMS I .
1016 0072 5144 XCHK22 /CHECK FOR APT SYSTEM.
1017 4473 PPRINT=JMS I .
1018 0073 5244 XPPRINT /PRINT A MESSAGE; FORM: "PRINT; MESSAGE"
1019 4474 READ=JMS I .
1020 0074 3416 XREAD
1021 4475 READCOMPARE=JMS I .
1022 0075 3412 XREADCOMPARE
1023 4476 RST=JMS I .
1024 0076 2441 XRST /STATUS AFTER "RST" COMMAND (12) / 5 TIMES 2
1025 4477 RSTB=JMS I .
1026 0077 2400 XNSTB /STATUS AFTER READ B-CODES COMMAND (16) 7 X 2

1027 4500 SETUP=JMS I .
1028 0100 4103 XSETUP
1029 4501 SPECIALTYPE=JMS I .
1030 0101 5352 XSPECIALTYPE
1031 4502 SCOPE=JMS I .
1032 0102 1333 XSCOPE
1033 4503 SUBSCOPE=JMS I .
1034 0103 1344 XSSCOPE
    
```

```

1035 4504 TAB=JMS I .
1036 0104 5215 XTAB / FORM: "TAB; N "
1037 4505 TY4OCT=JMS I .
1038 0105 5000 XTY4OCT /TYPE (4) OCTAL ; FORM: "TY4OCT; OCTAL "
1039 4506 TYPEIT=JMS I .
1040 0106 5305 XTYPEIT /TYPE 1 8 BIT ASCII, AC=ASCII.
1041 4507 WAITTY=JMS I .
1042 0107 5412 XWAITTY
1043 4510 WRITE=JMS I .
1044 0110 3200 XWRITE

1045 /
1046 /ACCUMULATOR SWITCH REGISTER DEFINITIONS
1047 /
1048 4000 SW0=4000
1049 2000 SW1=2000
1050 1000 SW2=1000
1051 0400 SW3=400
1052 0200 SW4=200
1053 0100 SW5=100
1054 0040 SW6=40
1055 0020 SW7=20
1056 0010 SW8=10
1057 0004 SW9=4
1058 0002 SW10=2
1059 0001 SW11=1

1060 /OPERATING SYSTEM ALLOCATED STORAGE REFERENCES
1061 /
1062 0111 0000 BUSY, 0 / = 1 - PROCESSING AN RX01 PROGRAM INTERRUPT
1063 0112 0000 COMMAND, 0 /DISKETTE COMMAND ; (AC) AT LCD
1064 0113 0000 COMPERROR, 0 /PROGRAM DATA COMPARE ERRORS
1065 0114 0000 DTESTP, 0 /DIAGNOSTIC TEST PARAMETERS (SELECTED AT L/S)
1066 0115 7777 FIRSTERROR, 7777 / (7777) IF 1ST ERROR ; (0) IF NOT
1067 0116 0000 HANGER, 0 /COUNTER TO DETECT DEVICE TEST HUNG
1068 0117 7777 K7777, -1 /
1069 0120 0000 RDC, 0 / = 0 IF A RDC TEST, = 7777 IF NOT
1070 0121 7765 KRETRY, -13 / 1 ORIGINAL TRY + 10 RETRYs
1071 0122 7746 SECTOPS, -32 / NEGATIVE # OF SECTORS PER TRACK (1-32 OCTAL)
1072 0123 0000 SSTART, 0 /SECTOR LAST ACCESSED ( 0 = "HOME" )
1073 0124 0000 STARGET, 0 /TARGET SECTOR OF (UNITX)
1074 0125 0000 START, 0 /TRACK LAST ACCESSED
1075 0126 0000 ASTATUS, 0 /DISKETTE STATUS AT ERROR OR DONE
1076 0127 0000 BSTATUS, 0 /RX01 DEFINITIVE ERROR CODE REGISTER
1077 0130 0000 CSTATUS, 0 /STATUS FROM THE "READ STATUS" COMMAND
1078 0131 0000 TARGET, 0 /TARGET TRACK OF (UNITX)
1079 0132 0000 TESTP, 0 /TEST PARAMETERS (DYNAMIC BY PROGRAM)
1080 0133 0000 XA10, 0
1081 0134 0000 XA11, 0
1082 0135 0000 UNITCK, 0 /VT78/AC11 (0)=RXA (1)=RXB
1083 0136 7677 MASK, 7677 /VT78/ 7677 FOR STANDARD 8 - 277 FOR VT78

1084 /
1085 /PROGRAM LOCATION XXX IS A TEMPORARY STORAGE REGISTER FOR DATA
1086 /OR ADDRESSES OF DATA WHICH ARE CALLED WITHIN SUBROUTINES WHICH
1087 /DO NOT CALL SUBROUTINES WHICH CALL THESE STORAGE REGISTERS
1088 /
1089 /XXX IS CALLED WITHIN THE FOLLOWING SUBROUTINES
    
```

```

1090 /
1091 /D,D/RX-ERROR (2)
1092 /COMPARE (2)
1093 /XGETAPATTERN (6)
1094 /RDORWR (2)
1095 /
1096 0137 0000 XXX, 0
1097 /
1098 0140 7765 R1RETRY, -13 /WERRORS /WRITE ERROR RECOVERY
1099 0141 7765 R2RETRY, -13 /RERRORS /READ ERROR RECOVERY
1100 0142 0000 DWSLOG, 0 /DATAERRORS /DATA ERROR WITH CRC STATUS ERROR
1101 0143 0000 DNSLOG, 0 /DNSERRORS /DATA ERROR BUT NO CRC STATUS ERROR
1102 0144 0000 SMDLOG, 0 /SMDERRORS /CRC STATUS ERROR BUT NO DATA ERROR
1103 0145 7765 SRETRY, -13 /SERRORS /SEEK ERROR RECOVERY
1104 0146 7765 PRETRY, -13 /PERERRORS /PARITY ERROR RECOVERY
1105 / (TRACKS) ARE SET TO THE NEGATIVE DIFFERENCE BETWEEN (OD), AND (ID)
1106 / IN THE SUBROUTINE "INITTRACKS "
1107 /
1108 0147 7663 TRACKS, -115 / -# OF TRACKS PER DISKETTE (-115 TO -1 DYNAMIC DECREMENT)
1109 0150 0115 TTRACKS, 115 / # OF TRACKS PER DISKETTE (115 TO 1 STATIC)
1110 0151 0000 XTARGET, 0 / ; (AC) = TRACK+SECTOR AT IOT LCD-B ; DESTINATION
1111 0152 0000 ECOMMAND, 0
1112 0153 0000 H1, 0
1113 0154 0000 GOBIT, 0 / > 0 MEANS EXPECTING AN RX01 PI, <= 0 MEANS NOT EXPECTING
1114 /
1115 / (PAT-SUMCHECK) IS A NUMBER GENERATED WITHIN SUBROUTINE " XGETAPATTERN"
1116 /EQUIVALENT TO SUMCHECK OF 60/124 [ 12/8 BIT MODE] DATA WORDS
1117 /
1118 /
1119 0155 0000 PATSUMCHECK, 0
1120 /
1121 / (WORDX) IS AN ADDRESS WHOSE CONTENTS = ; (PAT-SUMCHECK) + (WBUFFER) + (WBUFFER+1)
1122 /
1123 / (WORDY) IS AN ADDRESS FOR WHICH THE CONTENTS REPRESENT THE NEGATIVE-1
1124 /OF 2 TIMES THE CONTENTS OF THE ADDRESS WITHIN PROGRAM LOCATION " WORDX "
1125 /
1126 /THE CONTENTS OF BOTH WORDX AND WORDY ARE GENERATED WITHIN SUBROUTINE " XGETASECTOR "
1127 /
1128 0156 0000 WORDX, 0
1129 0157 0000 WORDY, 0
1130 0160 0000 LSB, 0
1131 0161 0000 MSB, 0
1132 0162 0000 TTYBUSY, 0
1133 0163 0000 PASS, 0
1134 0164 0000 0 /PASS COUNT TO A MAGNITUDE OF 16777215(10)
1135 0165 0000 RXHERE, 0 / = 7000 IF AN RX01 IS NOT CABLED TO THE RX8
1136 0166 0000 GOOD, 0 /EXPECTED RESULT
1137 0167 0000 EAC, 0 / (AC) AT ERROR "BAD" (ACTUAL RESULT)
1138 0170 0000 BLANK, 0
1139 0171 0600 TEST, 0 /ADDRESS OF STARTING ADDRESS OF TEST
1140 /*****
1141 /ADDITIONAL CALL FOR CONSOLE PACKAGE
1142 /
1143 /
1144 4572 CKSWIT= JMS I .

```

```

1145 0172 3400 XCKSWIT /CHECK IF LAS TO USE LOC 22 OR HARDWARE
1146 0173 7000 K7000, 7000
1147 0174 0007 K0007, 0007
1148 0175 7777 XCNT, -1
1149 4576 FORCE=JMS I .
1150 0176 5525 XXFORCE
1151 /STARTING ADDRESS 200 - (AC) = STARTUP SWITCHES
1152 /
1153 /STARTING ADDRESS 201 - RESTART (PARAMETERS ALREADY SELECTED AT START 200)
1154 /
1155 /
1156 0200 *200
1157 /
1158 /*****
1159 /CONSOLE
1160 /*****
1161 0200 5202 C8START, JMP ,+2 /NORMAL PROGRAM ACTIVITY,
1162 0201 5203 /RESTART WITH SAME PARAMETERS.
1163 /THIS SECTION IS NORMAL PROGRAM ACTIVITY.
1164 /DEVICE CODE 75 IS ASSUMED, ANY OTHER WILL CAUSE ERRORS.
1165 0202 7240 STA
1166 0203 3010 DCA A10
1167 0204 3162 DCA TTYBUSY
1168 0205 3111 DCA BUSY
1169 0206 3154 DCA GOBIT
1170 0207 3163 DCA PASS
1171 0210 3164 DCA PASS+1
1172 0211 6211 CDF 10 /CHANGE TO DATA FIELD OF CONSOLE PACKAGE
1173 0212 3777 DCA PASCNT /CLEAR CONSOLE PASS COUNTER
1174 0213 6201 CDF 0 /CHANGE BACK TO PROGRAM DATA FIELD
1175 0214 1376 TAD (-40)
1176 0215 3116 DCA HANGER
1177 0216 3123 DCA SSTART
1178 0217 3131 DCA TARGET
1179 0220 1010 TAD A10
1180 0221 7650 SNA CLA
1181 0222 5233 JMP AROUND /USE EXISTING PARAMETERS.
1182 /
1183 /*****
1184 0223 4775 JMS PNTID /PRINT ID AND REMOVE DIAGNOSTIC
1185 /*****
1186 0224 4424 CHECKC8 /CONSOLE ACTIVE
1187 0225 4425 XC8ENTR /ASK SR QUESTION,
1188 0226 5230 JMP ,+2
1189 0227 4461 HLT
1190 0230 4572 LAS /GET PARAMETERS.
1191 0231 3114 DCA DTESTP
1192 0232 4774 JMS CHNDEV /CHANGE DEVICE CODES.
1193 0233 1114 AROUND, TAD DTESTP /GET PARAMETERS
1194 0234 7040 CWA
1195 0235 0373 AND (7000 /MASK BITS 0-2
1196 0236 7650 SNA CLA
1197 0237 1373 TAD (7000
1198 0240 3165 DCA RXHERE/ = 0 IF RX01 CABLED TO RX8
1199 0241 1114 TAD DTESTP

```

```

1200 0742 7040 CMA
1201 0743 0372 AND (SW0+SW1)
1202 0744 3771 DCA UNITS / = UNITS(DRIVES) TO TEST
1203 0745 4770 JMS SELUNT /VT78/SELECT WHICH FLOPPY TO TEST
1204 0746 4473 PRINT
1205 0747 6533 MDETESTP /TEST PARAMETER CONFIRMATION MESSAGE
1206 0750 4505 TY4OCT
1207 0751 0114 DTESTP
1208
/ THE PROGRAM WILL VERIFY THAT THE CONTENTS OF PROGRAM LOCATIONS:
1209 /
1210 / OD, ID, FIRST, AND LAST
1211 /
/ WHICH ARE VARIABLE BY THE USER ARE WITHIN SELECTABLE LIMITS
1212 /
1213 /
1214 / 0 <= OD <= 114
1215 /
1216 0752 1027 TAD OD
1217 0753 7700 SMA CLA
1218 0754 5257 JMP ,+3
1219 0755 1367 TAD (52)
1220 0756 3027 DCA OD
1221 0757 1366 TAD (-114)
1222 0760 1027 TAD OD
1223 0761 7740 SMA SZA CLA
1224 0762 5255 JMP ,=5
1225 0763 4473 PRINT
1226 0764 6674 MOD
1227 0765 4505 TY4OCT
1228 0766 0027 UD
1229 / 0 <= ID <= OD
1230 /
1231 0767 1030 TAD ID
1232 0770 7700 SMA CLA
1233 0771 5274 JMP ,+3
1234 0772 1365 TAD (53)
1235 0773 3030 DCA ID
1236 0774 1030 TAD ID
1237 0775 7041 CIA
1238 0776 1027 TAD OD
1239 0777 7740 SMA SZA CLA
1240 0700 5272 JMP ,=6
1241 0701 4473 PRINT
1242 0702 6700 MID
1243 0703 4505 TY4OCT
1244 0704 0030 ID
1245 / 0 < (FIRST) <= 32
1246 /
1247 0705 1031 TAD FIRST
1248 0706 7740 SMA SZA CLA
1249 0707 5312 JMP ,+3
1250 0710 7301 CLL CLA IAC
1251 0711 3031 DCA FIRST
1252 0712 1031 TAD FIRST
1253 0713 1364 TAD (-32)
1254 0714 7740 SMA SZA CLA

```

```

1255 0315 5310 JMP ,=5
1256 0316 4473 PRINT
1257 0317 6704 MFIRST
1258 0320 4505 TY4OCT
1259 0321 0031 FIRST
1260 0322 5763 JMP TADLAST
1261
/IF THE TELEPRINTER IS BUSY (TTYBUSY = X), AND
1262 /
/IF A KEYBOARD FLAG HAS OCCURED,
1263 /
/THEN STOP TELEPRINTER OUTPUTS (IF A KRB = <CTRL>S ), OR
1264 /
1265 /THEN RESUME TELEPRINTER OUTPUTS (IF A KRB = <CTRL>Q
1266 /AND A PREVIOUS <CTRL> Q HAD OCCURED)
1267 /
1268
1269
1270 0323 6036 XKCC, KPB /READ THE KEYBOARD BUFFER STATIC
1271 0324 0362 AND (177
1272 0325 1361 TAD (200
1273 0326 6211 CDF 10 /MAKE IT 8 BIT CODE
1274 0327 3750 DCA I XC8CHAR /STORE IN FIELD ONE CHAR
1275 0330 6201 CDF 0
1276 0331 1162 TAD TTYBUSY /=1 IF BUSY
1277 0332 7012 RTR
1278 0333 7710 SPA CLA /WAS TELEPRINTER BUSY?
1279 0334 7040 CMA /YES
1280 0335 3351 DCA INMODE /0=NOT BUSY 7777=BUSY
1281 0336 3162 DCA TTYBUSY /CLEAR FLAG
1282 /*****
1283 /CONSOLE
1284 /;*****
1285
1286 0337 4424 C8TEST, CHECKC8 /CONSOLE ACTIVE.
1287 0340 4425 XC8ENTR /CHECK CONSOLE CONTROL CHARACTERS
1288 0341 7000 NOP
1289 0342 7200 CLA
1290 0343 1351 TAD INMODE
1291 0344 7650 SMA CLA /SKIP TO RESUME TYPEOUT
1292 0345 5760 JMP /EXIT
1293 0346 3351 DCA INMODE /CLEAR ACTIVE MESSAGE FLAG
1294 0347 5757 JMP /RESUME TYPEOUT
1295 0350 0031 XC8CHAR, C8CHAR /LOC IN FIELD 1
1296 0351 0000 INMODE, 0 /MESSAGE ACTIVE FLAG
1297
1298
/*****
1299 /CONSOLE
1300 /;*****
1301 0357 5344
1302 0360 5510
1303 0361 0200
1304 0362 0177
1305 0363 3273
1306 0364 7746
1307 0365 0053
1308 0366 7664
1309 0367 0052

```

1310 0770 6127
1311 0371 4235
1312 0372 6000
1313 0373 7000
1314 0374 5067
1315 0375 1531
1316 0376 7740
1317 0377 0661
0400

```

PAGE
/PRETEST = INITIALIZE [KEY] PART I / FLAG DETECTION PART I
/
/ (A) IF AN RX01 MICROCONTROLLER IS [NOT] CABLED TO THE RX8 INTERFACE,
/ THEN ALL FLAGS (DONE, TRANSFER REQUEST, AND ERROR) , AND THE RX8
/ INTERFACE TRANSFER REGISTER SHOULD HAVE BEEN CLEARED BY "KEY"
/ INITIALIZE (IF THEY WERE EVER SET).
/
/ (B) IF AN RX01 MICROCONTROLLER [IS] CABLED TO THE RX8 INTERFACE,
/ THEN "KEY" INITIALIZE SHOULD HAVE [SET] THE DONE FLAG BECAUSE
/ ANY [INIT] OF THE RX01 MICROCONTROLLER IS AN IMPLIED (READ SECTOR)
/ OF TRACK 0 SECTOR 1 (FOR SYSTEMS PROGRAMMING BOOTSTRAP APPLICATIONS).
/
/ THEREFORE, ANY ERROR (EXCEPT PARITY) THAT MAY OCCUR FROM A NORMAL
/ "READ SECTOR" COMMAND MAY OCCUR HERE CAUSEING THE ERROR FLAG TO SET, AND
/ DISPLAYING THE ERROR STATUS WITHIN THE TRANSFER REGISTER AT "DONE".
/
/ THE TRANSFER REQUEST FLAG SHOULD BE CLEARED.
/
/NOTE:
/
/SCOPE LOOPING IS NOT OFFERED BECAUSE THE "INIT" FUNCTION
/
PRETEST, STA
DCA FIRSTERROR /FOR FIRST ERROR EVER THIS PASS
DCA ERRORS /CLEAR "ERRORS" FOR FIRST "SCOPE" EVER
SCOPE /THIS "SCOPE" TO REFRESH "FAT" IF ERROR
TAD PCSCOPE
DCA TEST /TO REFRESH "FAT" FOR "ERROR"
CAF /VT78/
CKUNIT /VT78/SETUP FOR UNIT A OR B
STR
OK
EOPRE, ERROR /UNEXPECTED TRANSFER REQUEST FLAG
SER
OK
E2PRE, ERROR /UNEXPECTED ERROR FLAG
/
/*****
/*****
/IF AN RX01 MICROCONTROLLER [IS] CABLED TO THE RX8 INTERFACE
/THEN THE DONE FLAG SHOULD BE SET
/
TAD RXHERE
SZA CLA
JMP NORX01 /

```

1364 0421 4471
1365 0422 4445
1366 0423 5221
1367 0424 7410
1368 0425 4452
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379 0426 5775
1380
1381
1382 0427 1374
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396 0430 7107
1397 0431 3166
1398 0432 4441
1399 0433 3170
1400 0434 1170
1401 0435 0773
1402 0436 3167
1403 0437 1167
1404 0440 7041
1405 0441 1166
1406 0442 7650
1407 0443 5246
1408 0444 1167
1409 0445 4452
1410
1411
1412
1413
1414
1415
1416
1417
1418

```

WAIT
SDN
JMP ,-2 /WAIT FOR DONE FLAG
SKP
E3PRE, ERROR /MISSING DONE FLAG
/
/*****
/*****
/THE ENTIRE STATUS WORD IS DISPLAYED IN THE TRANSFER REGISTER AT ERROR/DONE TIME,
/
/IF AN RX01 CONTROLLER IS CABLED TO THE RX8 INTERFACE
/ (AND DRIVE 0 IS READY THEN THE STATUS SHOULD INDICATE " SEL DRV RDY" ), ALSO
/ DELETED DATA [MAY] = 1 IF TRACK 0/SECTOR 1 WAS WRITTEN WITH DELETED DATA
/ AND "INIT DONE" SHOULD BE SET.
/
JMP TSTUNT /OFF PAGE BECAUSE OF ROOM.
/
TAD (40 /PROGRAM EXPECTS DRIVE 0 TO BE READY
/
////////////////////////////////////////////////////
//
// 4 5 - - 8 9 10 11 //
//
// SEL WRITE INIT PAR //
// DRIVE DD PROTECT [DONE] CRC //
// RDY (N/A) //
//
////////////////////////////////////////////////////
/
CLL IAC RTL / 4 [INIT] DONE OR 204
NORX01, DCA GOOD
XDRIN
DCA BLANK /ACTUAL STATUS FROM [INIT]
TAD BLANK
AND COMP
DCA EAC /STATUS MINUS DELETED DATA (BIT 5)
TAD EAC
CIA
TAD GOOD /EXPECTED
SMA CLA
JMP ,+3 /OK
TAD EAC
E1PRE, ERROR / [INIT] STATUS NOT = EXPECTED
/
/IF AN RX01 MICROCONTROLLER [IS] CABLED TO THE RX8 INTERFACE
/THEN THE PREVIOUS "SDN" SHOULD HAVE CLEARED THE DONE FLAG, BUT
/
/IF AN RX01 MICROCONTROLLER IS [NOT] CABLED TO THE RX8 INTERFACE
/THEN "KEY" INITIALIZE SHOULD HAVE CLEARED THE DONE FLAG
/
/TECHNICAL NOTE:
/

```

```

1419 /IF THE DONE FLAG IS SET, AND IF THE INTERRUPT ENABLE FLIP-FLOP IS SET ILLEGALLY,
1420 /THEN AN "UNEXPECTED RX01 INTERRUPT" WILL OCCUR IN TO (IF AN RX01 CONTROLLER
1421 / (IS) CABLED TO THE RX8 INTERFACE) OR IN T1 WHEN THE MAINTENANCE FLIP-
1422 /FLOP "SETS ALL FLAGS"
1423 /
1424 0446 4445 SDN
1425 0447 4465 OK
1426 0450 4452 E4PRE, ERROR /UNEXPECTED DONE FLAG
1427 /
1428 /END OF PRE-TEST /END OF PRETEST
1429 /
1430 0451 5311 JMP REBEGIN
1431 /
1432 0452 4464 MGRETESTS, LOCKUP
1433 0453 3777* FIRSTTEST, DCA ERRORS
1434 0454 1413 TAD I A13
1435 0455 3171 DCA TEST / FAT (FIRST ADDRESS OF TEST)
1436 0456 1171 TAD TEST
1437 0457 3776* DCA PCSCOPE / EQUIVALENT TO " SCOPE "
1438 0460 3772* DCA WUNITS /FOR FIRST ENTRY INTO XGETUNIT THIS TEST
1439 0461 5571 JMP I TEST
1440 /
1441 /THERE ARE NO MORE TESTS
1442 /
1443 /PRINT AN END OF PASS INDICATOR
1444 /
1445 / A - INTERFACE TEST OK (ONLY RX8 TO TEST)
1446 / C - RX8 AND RX01 TEST OK
1447 / D - RX8 AND RX01 AND DRIVE TESTING OK
1448 /
1449 / - - AN ERROR OCCURED (DURING A, B, OR D)
1450 /
1451 0400 XD=0400
1452 0462 2771* ISZ CHECKU /VT78/ARE WE DONE ALL SELECTED UNITS?
1453 0463 5323 JMP NXTUNT /VT78/NO - DO NEXT UNIT(RXB)
1454 0464 1370 TAD (XD)
1455 0465 3322 NOMGRETESTS, DCA MX / (X1), (XC), (XD), OR 0
1456 0466 4767* JMS SELUNT /RESET UNIT COUNTER(CHECKU)
1457 /
1458 /NOTE:IF THE CONTENTS OF PROGRAM LOCATION FIRSTERROR = 0
1459 /THEN AN ERROR HAS OCCURED FOR THIS PASS
1460 /
1461 0467 1115 TAD FIRSTERROR
1462 0470 7640 SZA CLA
1463 0471 5274 JMP +3
1464 0472 1366 TAD (5500)
1465 0473 3322 DCA MX / -
1466 /*****
1467 /CONSOLE
1468 /*****
1469 0474 4424 CHECKC8
1470 0475 4426 C8PASS
1471 0476 5326 JMP C8RET2 /
1472 /*****
1473 0477 4473 PRINT
    
```

```

1474 0400 0522 MX
1475 0401 2163 ISZ PASS
1476 0402 5305 JMP +3
1477 0403 2164 ISZ PASS+1
1478 0404 7000 NOP
1479 0405 4572 LAS
1480 0406 0370 AND (SW3)
1481 0407 7640 SZA CLA
1482 0410 4461 HLT
1483 0411 1114 REBEGIN, TAD DTESIP
1484 0412 0365 AND (37)
1485 0413 1364 TAD (TESTS-1)
1486 0414 3013 DCA A13
1487 0415 7240 STA
1488 0416 3115 DCA FIRSTERROR /FIRST ERROR SWITCH FOR EACH PASS
1489 0417 7340 CLL CLA CMA
1490 0420 3763* DCA CLKCNT /FOR APT TIMING
1491 0421 5253 JMP FIRSTTEST
1492 /
1493 0422 1100 MX, TEXT "1" / I, C, OR D
1494 /
1495 0423 7201 NXTUNT, CLA IAC
1496 0424 3135 DCA UNITCK /SET UP TO TEST UNIT B
1497 0425 5200 JMP PRETEST /START OVER
1498 /
1499 /*****
1500 /ROUTINE FOR CONSOLE PASS
1501 /
1502 0426 6001 C8RET2, ION /CONSOLE PASS
1503 0427 2163 ISZ PASS
1504 0430 5333 JMP +3
1505 0431 2164 ISZ PASS+1
1506 /
1507 0432 7000 NOP
1508 0433 5311 JMP REBEGIN /CONTINUE WITH PROGRAM RETURN
1509 /*****
1510 0463 4151
1511 0464 4727
1512 0465 0037
1513 0466 5500
1514 0467 6127
1515 0470 0400
1516 0471 6151
1517 0472 4236
1518 0473 2746
1519 0474 0040
1520 0475 2732
1521 0476 1366
1522 0477 1365
1523 PAGE
1524 /TEST 0 - FLAG DETECTION PART II / " C " LINES VERIFICATION PART I
1525 /
1526 /*****
1527 /
    
```

```

1528 /IF AN RX01 MICROCONTROLLER IS CABLED TO THE RX8 INTERFACE
1529 /THEN DON'T EXECUTE THIS TEST
1530 /BECAUSE ISSUING THE IOT LCD WITH THE AC = 177
1531 /RESEMBLED A COMMAND TO THE RX01
1532 /
1533 0600 1165 TO, TAD RXHERE
1534 0601 7650 SNA CLA
1535 0602 5453 NOTEST
1536 0603 3166 DCA GOOD
1537 /*****
1538 /*****
1539 /
1540 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT THE LCD (LOAD COMMAND REGISTER)
1541 /IOT 67X1 DOES [NOT] SET THE MAINTENANCE FLIP-FLOP WHEN THE CONTENTS
1542 /OF THE AC = 177 AT THE TIME THE LCD IOT IS ISSUED.
1543 /
1544 /
1545 /TECHNICAL NOTE:
1546 /
1547 /IF AN ERROR OCCURS, THEN IT IS ASSUMED [KEY] INIT FAILED TO CLEAR THE
1548 /MAINTENANCE FLIP-FLOP, OR, THAT THE ISSUING OF THE LCD IOT REALLY
1549 / [SET] THE MAINTENANCE FLIP-FLOP INSTEAD OF [CLEARING] .
1550 /
1551 / " C " LINES VERIFICATION PART I
1552 /
1553 0604 1377 TAD (177)
1554 0605 4436 LCD /MAINTENANCE MODE <OFF>
1555 /THE (AC) SHOULD = 0 BECAUSE IOT LCD 67X1 SHOULD CLEAR THE AC
1556 /
1557 0606 7440 SZA
1558 0607 4452 E0, ERROR / IOT 67X1 DID NOT CLEAR THE AC
1559 0610 4503 SUBSCOPE
1560 /
1561 /FLAG DETECTION PART II
1562 /
1563 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT ISSUING IOT LCD 67X1 WITH
1564 /THE AC = 177 DOES NOT SET THE MAINTENANCE FLIP-FLOP
1565 /WHICH IN TURN WOULD SET ALL FLAGS
1566 /
1567 /THEREFORE ALL FLAGS SHOULD BE CLEARED
1568 /
1569 0611 4443 STR
1570 0612 4465 OK
1571 0613 4452 E1, ERROR /UNEXPECTED TRANSFER REQUEST FLAG
1572 0614 4503 SUBSCOPE
1573 0615 4444 SER
1574 0616 4465 OK
1575 0617 4452 E2, ERROR /UNEXPECTED ERROR FLAG
1576 0620 4503 SUBSCOPE
1577 0621 4445 SDN
1578 0622 4465 OK
1579 0623 4452 E3, ERROR /UNEXPECTED DONE FLAG
1580 0624 4502 SCOPE
1581 0625 5453 EXIT / END OF TEST 0
1582 /
    
```

```

1583 /TEST 1 - DIRECTION OF IOT XDR (67X2) PART I / IOT DECODING PART I
1584 /
1585 / - " C " LINES VERIFICATION PART II
1586 /
1587 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT
1588 /ISSUING THE IOT XDR (TRANSFER DATA REGISTER) 67X2 DOES [NOT] CLEAR
1589 /THE MAINTENANCE FLIP-FLOP
1590 /
1591 /TECHNICAL NOTE:
1592 /
1593 /THE IOT'S SDN (67X5), AND SER (67X4) ARE NOT TESTED HERE
1594 /BECAUSE IF AN RX01 MICROCONTROLLER IS CABLED TO THE RX8 INTERFACE
1595 /AND IF THE IOT LCD IS ISSUED WITH THE AC = 200
1596 /REALLY CLEARS THE MAINTENANCE FLIP-FLOP
1597 /THEN THE DONE FLAG, AND THE ERROR FLAG SHOULD BE CLEARED, AND
1598 /TRANSFER REQUEST MAY BE SET
1599 /BECAUSE THE CLEARING OF THE MAINTENANCE FLIP-FLOP WOULD HAVE
1600 /RESEMBLED A COMMAND TO THE RX01 MICROCONTROLLER
1601 /
1602 /TECHNICAL NOTE:
1603 /
1604 /IF THE CONTENTS OF THE TRANSFER REGISTER IS NOT = 200, THEN IS MUST
1605 /BE ASSUMED THAT THE SECOND LCD IOT CLEARED THE MAINTENANCE FLIP-FLOP
1606 /OR THAT IOT XDR CLEARED THE MAINTENANCE FLIP-FLOP
1607 /
1608 0626 3166 T1, DCA GOOD
1609 0627 1376 TAD (200)
1610 0630 4436 LCD / MAINTENANCE MODE <ON>
1611 /THE (AC) SHOULD = 0 AFTER ISSUING IOT LCD 67X1
1612 /
1613 0631 7440 SZA
1614 0632 4452 E11, ERROR / IOT LCD 67X1 FAILED TO CLEAR AC
1615 0633 4503 SUBSCOPE
1616 /
1617 0634 1376 TAD (200)
1618 0635 4436 LCD / MAINTENANCE MODE <ON>, AGAIN
1619 0636 4441 XDRIN /CONTENTS OF TRANSFER REGISTER
1620 0637 3167 DCA EAC /SAVE
1621 0640 1167 TAD EAC
1622 0641 1375 TAD (-200) /COMPARE WITH "EXPECTED"
1623 0642 7650 SNA CLA
1624 0643 5250 JMP ,+5 / OK
1625 0644 1376 TAD (200)
1626 0645 3166 DCA GOOD / "EXPECTED" RESULT
1627 0646 1167 TAD EAC / "ACTUAL" RESULT
1628 0647 4452 E10, ERROR /TRANSFER REGISTER NOT #200
1629 0650 4502 SCOPE
1630 0651 5453 EXIT / END OF TEST 1
1631 /TEST 2 - FLAG DETECTION PART III / " C " LINES VERIFICATION PART III
1632 /
1633 / (A) THE SETTING OF THE MAINTENANCE FLIP-FLOP SHOULD "DIRECT SET" ALL
1634 / FLAGS (DONE, TRANSFER REQUEST, AND ERROR).
1635 /
1636 / (B) IF AN RX01 MICROCONTROLLER [IS] CABLED TO THE RX8 INTERFACE,
1637 / THEN THE SETTING OF THE MAINTENANCE FLIP-FLOP WILL ASSERT THE "RUN"
    
```

```

1638 / LINE (RESEMBLING A FILL BUFFER COMMAND) THUS CAUSING THE RX01
1639 / CONTROLLER TO SETUP FOR A "FILL BUFFER", BUT, HOWEVER, BECAUSE THE
1640 / MAINTENANCE FLIP-FLOP (IS) SET, THE RX8 INTERFACE RECIEVERS
1641 / SHOULD BE DISABLED AND NOT REACTIVE TO THE RX01 MICROCONTROLLER.
1642 /
1643 /WITH ALL FLAGS SET, THE RX8 INTERFACE IOT'S;
1644 /
1645 / SDN = "SKIP ON DONE" (67X5), AND
1646 / SER = "SKIP ON ERROR" (67X4), AND
1647 / STR = "SKIP ON TRANSFER REQUEST" (67X3) SHOULD SKIP
1648 /
1649 /TECHNICAL NOTE:
1650 /
1651 /IF FLAGS ARE "MISSING", IS THE MAINTENANCE MODE FLIP-FLOP REALLY SET ?
1652 /
1653 0652 1376 T2, TAD (200)
1654 0653 4436 LCD / MAINTENANCE <ON>
1655 0654 4445 SDN
1656 0655 4452 E20, ERROR /MISSING DONE FLAG
1657 0656 4503 SUBSCOPE
1658 0657 4443 STR
1659 0660 4452 E21, ERROR /MISSING TRANSFER REQUEST FLAG
1660 0661 4503 SUBSCOPE
1661 0662 4444 SER
1662 0663 4452 E22, ERROR /MISSING ERROR FLAG
1663 0664 4503 SUBSCOPE
1664 /
1665 /ALL FLAGS SHOULD REMAIN " DIRECT SET "
1666 /BECAUSE THE MAINTENANCE FLIP-FLOP SHOULD STILL BE SET.
1667 /
1668 /TECHNICAL NOTE:
1669 /
1670 /IF THE FLAGS ARE "MISSING" THEN IT IS ASSUMED THAT THE PREVIOUS
1671 /FLAG TESTING ACTUALLY (CLEARED) THE FLAGS.
1672 /
1673 0665 4445 SDN
1674 0666 4452 E23, ERROR /MISSING DONE FLAG
1675 0667 4503 SUBSCOPE
1676 0670 4443 STR
1677 0671 4452 E24, ERROR /MISSING TRANSFER REQUEST FLAG
1678 0672 4503 SUBSCOPE
1679 0673 4444 SER
1680 0674 4452 E25, ERROR /MISSING ERROR FLAG
1681 0675 4503 SUBSCOPE
1682 / " C " LINES VERIFICATION PART III
1683 /
1684 /THE FOLLOWING RX8 INTERFACE IOT'S SHOULD NOT CLEAR THE AC;
1685 /
1686 /IOT'S: SDN(67X5), SER(67X4), OR STR(67X3)
1687 /
1688 0476 7240 STA
1689 0477 3166 DCA GOOD
1690 0700 1166 TAD GOOD
1691 0701 6755 K67X5B, 6755
1692 0702 7000 NOP

```

```

1693 0703 7650 SNA CLA
1694 0704 4452 E26, ERROR / IOT SDN (67X5) CLEARED THE AC
1695 0705 4503 SUBSCOPE
1696 0706 1166 TAD GOOD
1697 0707 6753 K67X3B, 6753
1698 0710 7000 NOP
1699 0711 7650 SNA CLA
1700 0712 4452 E27, ERROR / IOT STR (67X3) CLEARED THE AC
1701 0713 4503 SUBSCOPE
1702 0714 1166 TAD GOOD
1703 0715 6754 K67X4B, 6754
1704 0716 7000 NOP
1705 0717 7650 SNA CLA
1706 0720 4452 E28, ERROR / IOT SER (67X4) CLEARED THE AC
1707 0721 4502 SCOPE
1708 0722 5453 EXIT / END OF TEST 2
1709 /
1710 /TEST 3
1711 /
1712 /RX8 IOT DEVICE CODE VERIFICATION
1713 /
1714 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT ONLY THE DEVICE CODE SELECTED
1715 /BY THE OPERATOR (AC SWITCHES 3-4-5 AT THE START OF THIS PROGRAM) IS ACTIVE.
1716 /
1717 /FIRST SET THE MAINTENANCE FLIP-FLOP, WHICH HAS PREVIOUSLY BEEN VERIFIED TO
1718 /DIRECT SET ALL FLAGS, THEN SEQUENCE THROUGH ALL DEVICE CODES (EXPECT THE
1719 /DEVICE CODE SELECTED AT THE START OF THIS PROGRAM) BY ISSUING IOT SDN 67X5
1720 / (SKIP ON DONE FLAG), WHICH HAS ALSO PREVIOUSLY BEEN VERIFIED TO "SKIP AND
1721 /CLEAR " SUCCESSFULLY.
1722 /
1723 /NOTE:
1724 /
1725 /THE PROGRAM DOES NOT ISSUE THE DEVICE CODE 67X5 WHERE X = POSITION OF AC
1726 /SWITCHES 3-4-5 AT THE START 200 OF THIS PROGRAM.
1727 /*****
1728 /THIS TEST HAS BEEN REMOVED
1729 /THIS TEST IS NOT NEEDED IN A SYSTEM ENVIREMENT.
1730 /TO REPLACE TEST T3 CHANGE LOCATIONS:
1731 / LOCATION FROM TO
1732 / -----
1733 / 734 5464 1367
1734 /
1735 / 767 XXXX 0200
1736 /REMOVED MAY 16,1975
1737 ///
1738 /
1739 0723 5453 T3, EXIT
1740 0724 4436 LCD / MAINTENANCE <ON>
1741 0725 1774 TAD K67XA
1742 0726 3166 DCA GOOD
1743 0727 1373 TAD (=7)
1744 0730 3010 DCA A10
1745 0731 1372 TAD (6705)
1746 0732 3341 DCACTIVE, DCA ACTIVE
1747 0733 4502 SCOPE /REFRESH PROGRAM LOCATION PCSCOPE

```

```

1748 0734 1166 TAD GOOD
1749 0735 7041 CIA
1750 0736 1341 TAD ACTIVE
1751 0737 7650 SNA CLA
1752 0740 5346 JMP NEXACTIVE
1753 0741 6775 ACTIVE, 6775
1754 0742 5346 JMP NEXACTIVE
1755 0743 1341 TAD ACTIVE
1756 0744 4452 E30, ERROR
1757 0745 4502 SCOPE / (AC) = ILLEGAL DEVICE CODE
1758 0746 1371 NEXACTIVE, TAD (10)
1759 0747 1341 TAD ACTIVE
1760 0750 2010 ISZ A10
1761 0751 5332 JMP DCACTIVE
1762 0752 5453 EXIT / END OF TEST 3
1763 0771 0010
1764 0772 6705
1765 0773 7771
1766 0774 6422
1767 0775 7600
1768 0776 0200
1769 0777 0177
1770 1000

```

```

PAGE
/TEST 4 - TRANSFER REGISTER DIRECTION TESTING (PART II)
/
/ - " C " LINES VERIFICATION PART IV
/
/ WITH THE MAINTENANCE FLIP-FLOP SET THE PROGRAM WILL VERIFY THE DIRECTION
/ AND TRANSFER MODE (8-BIT MODE INCLUSIVE "OR", AND 12-BIT MODE "JAM")
/ TRANSFERS INTO THE ACCUMULATOR FROM THE RX8 TRANSFER REGISTER BY ISSUING
/ IOT "XDR" (TRANSFER DATA REGISTER) 67X2 AFTER PREVIOUSLY [LOADING] THE
/ THE TRANSFER REGISTER WITH THE CONTENTS OF THE ACCUMULATOR REPRESENT-
/ ATIVE OF THE FLOWING PATTERNS WHEN THE "LCD" IOT 67X1 IS ISSUED.
/
/ (1) 200 - MAINTENANCE MODE <ON>
/ (2) 376 -
/ (3) 375 -
/ (4) 373 - (BYTES 2 THRU 7)
/ (5) 367 - (INCLUSIVE "OR" )
/ (6) 357 -
/ (7) 337 -
/ (8) 7677 - (WORD 8 - "JAM" )
/
/ THE LCD IOT WILL BE ISSUED A TOTAL OF 8 TIMES.
/
/ THE 1ST LCD IOT WILL BE ISSUED WITH THE AC = 200 WHICH INITIALLY SETS THE
/ MAINTENANCE FLIP-FLOP THEREBY GUARANTEEING THE CONTENTS OF THE TRANSFER
/ REGISTER [ADTER] EACH SUCCEEDING LCD IOT.
/
/ LCD IOT'S 2 THUR 8 APE ISSUED WITH THE ACCUMULATOR CONTAINING THE PATTERNS
/ DESCRIBED ABOVE.
/
/ ALL PATTERNS EXCEPT WORD 8 (7677) TEST THE INCLUSIVE "OR" TRANSFER OF
/ THE RX8 INTERFACE TRANSFER REGISTER. WORD 8 TESTS THE 12-BIT "JAM" TRANSFER.
/

```

```

1802 1000 1377 T4, TAD (200)
1803 1001 4436 LCD /MAINTENANCE MODE <UN>
1804 1002 7350 CLL STA RAK / 3777
1805 1003 3170 DCA BLANK
1806 1004 1170 T48, TAD BLANK
1807 1005 7120 STL
1808 1006 7500 SMA
1809 1007 7100 CLL
1810 1010 7004 RAL
1811 1011 3170 DCA BLANK
1812 1012 4502 SCOPE / REFRESH PROGRAM LOCATION PCSCOPE
1813 1013 1170 TAD BLANK / (BLANK) = (AC) BEFORE LCD IOT 67X1
1814 1014 4436 LCD / TO
1815
1816 / " C " LINES VERIFICATION PART IV
1817 /
1818 / THE PURPOSE OF THIS TEST IS TO VERIFY THAT SUCCEEDING LCD IOT'S(67X1)
1819 / TRANSFER THE (AC) INTO THE DATA REGISTER CLEARING THE ACCUMULATOR
1820 /
1821 1015 3167 DCA EAC / (AC) AFTER ISSUING IOT LCD (67X1)
1822 1016 3166 DCA GOOD / PROGRAM EXPECTS AC = 0
1823 1017 1167 TAD EAC
1824 1020 7440 SZA
1825 1021 4452 E42, ERROR / IOT LCD (67X1) DIDN'T CLEAR THE AC
1826 1022 4503 SUBSCOPE
1827
1828 / TRANSFER DIRECTION PART II
1829 /
1830 1023 4441 XDRIN / FROM
1831 1024 3167 DCA EAC
1832 1025 1376 TAD (100)
1833 1026 0170 AND BLANK
1834 1027 7106 CLL RTL
1835 1030 7006 RTL / LINK = 1 FOR 8-BIT MODE
1836 1031 7006 RTL
1837 1032 1170 TAD BLANK
1838 1033 7430 SZL
1839 1034 0375 AND (377) / 8-BIT BYTE "GOOD" MASK
1840 1035 3166 DCA GOOD
1841 1036 1166 TAD GOOD /EXPECTED RESULT
1842 1037 7041 CIA
1843 1040 1167 TAD EAC /ACTUAL RESULT
1844 1041 7650 SNA CLA
1845 1042 5245 JMP +3 /COMPARED OK
1846 1043 1167 TAD EAC
1847 1044 4452 E40, ERROR / TRANSFER REGISTER NOT = "GOOD"
1848 /CHECK TO SEE IF RUNNING ON A VT78 SYSTEM
1849 1045 4365 JMS VT78CK /VT78/ RETURN +1 IF NOT VT78
1850 1046 5261 JMP E41+1 /VT78 = SKIP FOLLOWING SUBTEST
1851
1852 1047 4503 SUBSCOPE
1853 /
1854 /THE TRANSFER REGISTER SHOULD REMAIN UNCHANGED
1855 /FROM THE PREVIOUS XDR IOT (NOT SO ON VT78)
1856 /

```



```

1857 1050 4441 XDRIN / FROM
1858 1051 3167 DCA EAC
1859 1052 1167 TAD EAC /ACTUAL
1860 1053 7041 CIA
1861 1054 1166 TAD GOUD /EXPECTED
1862 1055 7650 SNA CLA
1863 1056 5261 JMP ,+3 /COMPARED OK
1864 1057 1167 TAD EAC
1865 1060 4452 E41, ERROR / TRANSFER REGISTER NOT = "GOOD"
1866 1061 4502 SCOPE
1867 1062 1170 TAD BLANK
1868 1063 0376 AND (100)
1869 1064 7640 SZA CLA
1870 1065 5204 JMP T4B /UNTIL (BLANK) = 7677
1871 1066 5453 EXIT / END OF TEST 4
1872 /
1873 /
1874 /RX8 IOT DECODING VERIFICATION PART II
1875 /
1876 /*****
1877 /*****
1878 /
1879 /IF AN RX01 CONTROL IS CABLED TO THE RX8 INTERFACE
1880 /THEN DON'T EXECUTE THIS TEST
1881 /BECAUSE THE CLEARING OF THE MAINTENANCE F/F
1882 /RESEMBLES A FILL BUFFER COMMAND (NOT SO ON VT78)
1883 /
1884 /CHECK TO SEE IF RUNNING ON A VT78 SYSTEM
1885 1067 4365 JMS VT78CK /VT78/ RETURN +1 IF NOT VT78
1886 1070 5274 JMP ,+4 /VT78 -ALWAYS EXECUTE TEST
1887 /
1888 1071 1165 I5, TAD RXHERE
1889 1072 7650 SNA CLA
1890 1073 5453 NOTEST
1891 1074 3166 DCA GOOD
1892 /*****
1893 /*****
1894 /
1895 /THE MAINTENANCE FLIP-FLOP HAS PREVIOUSLY BEEN VERIFIED TO SET AND CLEAR,
1896 /THE IOT UNDER TEST SHOULD "SKIP AND CLEAR" (ONLY) ITS RESPECTIVE FLAG,
1897 /ALL OTHER FLAGS SHOULD REMAIN UNCHANGED
1898 /
1899 / (I.E. THE SDN IUT 67X5 SHOULD SKIP AND CLEAR ONLY THE DONE FLAG, ALL
1900 /OTHER FLAGS SHOULD REMAIN SET)
1901 /
1902 1075 1377 TAD (200)
1903 1076 4436 LCD
1904 1077 4436 LCD / MAINTENANCE <ON> / <OFF>
1905 1100 4441 XDRIN
1906 1101 7440 SZA
1907 1102 4452 E56, ERROR /TRANSFER REGISTER NOT = 0
1908 1103 4503 SUBSCOPE
1909 1104 4445 SDN
1910 1105 4452 E50, ERROR /DONE FLAG WASN'T SET, OR
1911 1106 4503 SUBSCOPE /IOT LCD OR XDR CLEARED THE DONE FLAG

```

```

1912 1107 4445 SDN
1913 1110 4465 UK
1914 1111 4452 E53, ERROR
1915 1112 4503 SUBSCOPE /IOT SDN DIDN'T "SKIP AND CLEAR"
1916 1113 4443 STR
1917 1114 4452 E51, ERROR /TRANSFER REQUEST FLAG WASN'T EVER SET, OR
1918 1115 4503 SUBSCOPE /IOT LCD, OR SDN OR XDR CLEARED THE TR FLAG
1919 1116 4443 STR
1920 1117 4465 OK
1921 1120 4452 E54, ERROR
1922 1121 4503 SUBSCOPE /IOT STR DIDN'T "SKIP AND CLEAR"
1923 1122 4444 SER
1924 1123 4452 E52, ERROR /ERROR FLAG WASN'T EVER SET, OR
1925 1124 4503 SUBSCOPE /IOTS LCD OR SDN OR XDR OR STR CLEARED THE ERROR FLAG
1926 1125 4444 SER
1927 1126 4465 OK
1928 1127 4452 E55, ERROR
1929 1130 4502 SCOPE /IOT SER DIDN'T "SKIP AND CLEAR"
1930 1131 5453 EXIT / END OF TEST 5
1931 /
1932 /TEST 6 - INTERRUPT TEST PART I / IOT DECODING VERIFICATION PART III
1933 /
1934 /INTERRUPT TEST PART I
1935 /
1936 /THE MAINTENANCE FLIP-FLOP HAS PREVIOUSLY BEEN VERIFIED TO DIRECT
1937 / [SET] ALL FLAGS AND THE INTERFACE IUT -SKIP ON DONE= "SDN" 67X5 WAS
1938 /FOUND TO "SKIP AND CLEAR" SUCCESSFULLY,
1939 /
1940 /FIRST SET THE MAINTENANCE FLIP-FLOP WHICH IN TURN SETS ALL FLAGS,
1941 /
1942 /THEN ISSUE IOT INTR 67X6 WITH THE AC = 0 [CLEARING] THE RX8 INTERRUPT ENABLE
1943 /NO INTERRUPTS SHOULD OCCUR
1944 /
1945 1132 4507 T6, WAITTY
1946 1133 1374 TAD (E60)
1947 1134 3002 DCA IPI
1948 1135 1377 TAD (200)
1949 1136 4436 LCD
1950 1140 7000 INTR /INTERRUPT ENABLE FLIP-FLOP <OFF>
1951 /
1952 1141 7410 NOP /...WAIT
1953 1142 4452 E60, ERROR /...PLENTY
1954 1143 4503 SUBSCOPE /...OF TIME
1955 /
1956 /*****
1957 /*****
1958 /
1959 /IF AN RX01 MICRO-CONTROLLER IS CABLED TO THE RX8 INTERFACE
1960 /THEN DON'T EXECUTE THE REMAINING PORTION OF THIS TEST
1961 /BECAUSE THE CLEARING OF THE MAINTENANCE FLIP-FLOP RESEMBLES A FILL BUFFER COMMAND
1962 /TO THE RX01 MICROCONTROLLER (NOT SO ON VT78)
1963 /
1964 /CHECK TO SEE IF RUNNING ON A VT78 SYSTEM
1965 1144 4365 JMS VT78CK /VT78/ RETURN +1 IF NOT VT78
1966 1145 5351 JMP ,+4 /VT78 -ALWAYS EXECUTE TEST

```

```

1967
1968      1146 1165      TAD RXHERE
1969      1147 7650      SNA CLA
1970      1150 5453      NOTEST
1971      /
1972      /*****
1973      /*****
1974      /
1975      /IOT DECODING PART III- IOT INTR 67X6 DECODING VERIFICATION
1976      /
1977      /TECHNICAL NOTE:
1978      /
1979      /ALL FLAGS SHOULD REMAIN SET
1980      /IF ANY FLAG IS MISSING,
1981      /THEN IT IS ASSUMED THAT IOT "INTR" 67X6 CLEARED THE FLAG(S)
1982      /
1983      1151 4436      LCD      /MAINTENANCE MODE <OFF>
1984      1152 4446      INTR     /DISABLE THE INTERRUPT ENABLE F/F
1985      1153 4445      SDN
1986      1154 4452      E61,   ERROR      /MISSING DONE FLAG
1987      1155 4503      SUBSCOPE
1988      1156 4443      STR
1989      1157 4452      E62,   ERROR      /MISSING TRANSFER REQUEST FLAG
1990      1160 4503      SUBSCOPE
1991      1161 4444      SER
1992      1162 4452      E63,   ERROR      /MISSING ERROR FLAG
1993      1163 4502      SCOPE
1994      1164 5453      EXIT     / END OF TEST 6
1995      /ROUTINE TO CHECK IF RUNNING ON A VT78 SYSTEM
1996      /
1997      1165 0000      VI78CK, 0
1998      1166 1022      TAD 22      /GET LOC 22 (HCW2)
1999      1167 0373      AND (1000   /TEST BIT 2
2000      1170 7650      SNA CLA    /IS THIS A VT78 SYSTEM?
2001      1171 2365      ISZ VT78CK /NO - BUMP RETURN
2002      1172 5765      JMP I VT78CK /RETURN
2003
2004      1173 1000
2005      1174 1142
2006      1175 0377
2007      1176 0100
2008      1177 0200
2009      /
2010      /TEST 7 - INTERRUPT TEST PART II
2011      /
2012      /INTERRUPT TEST PART II
2013      /
2014      /THE MAINTENANCE FLIP-FLOP HAS PREVIOUSLY BEEN VERIFIED TO DIRECT
2015      / [SET] ALL FLAGS AND THE INTERFACE IOT -SKIP ON DONE- "SDN" 67X5 WAS
2016      / FOUND TO "SKIP AND CLEAR" SUCCESSFULLY,
2017      /
2018      /FIRST SET THE MAINTENANCE FLIP-FLOP
2019      /WHICH SHOULD DIRECT SET THE DONE FLAG,
2020      /THEN BY SETTING THE RX01 INTERRUPT ENABLE

```

```

2021      /BY ISSUING THE IOT "INTR" 67X6 WITH THE AC = 1.
2022      /
2023      /AN INTERRUPT REQUEST SHOULD BE ASSERTED.
2024      /
2025      /THE PROGRAM IS EXPECTING AN INTERRUPT.
2026      /
2027      /TECHNICAL NOTE:
2028      /
2029      /IF AN INTERRUPT DOES NOT OCCUR, THEN IT IS ASSUMED THAT ISSUING THE IOT
2030      / "INTR" 67X6 DID NOT SET THE RX8 INTERRUPT ENABLE, OR INTERRUPT REQUEST
2031      /
2032      1200 4507      T7,   WAITTY
2033      1201 1377      TAD (200)
2034      1202 4436      LCD      /MAINTENANCE <ON>
2035      1203 1376      TAD (T7OK)
2036      1204 3002      DCA IPI
2037      1205 6001      ION      /*SEE FOOTNOTE NEXT PAGE
2038      1206 7201      CLA IAC
2039      1207 4446      INTR     /RX01 INTERRUPT ENABLE <ON>
2040      1210 7000      NOP
2041      /
2042      /PROGRAM NOTE:
2043      /
2044      /CLEAR PROGRAM LOCATION "GOBIT" BECAUSE THE TIME FOR THE INTERRUPT
2045      / TO OCCUR HAS EXPIRED (IF IT WAS EVER GOING TO OCCUR THAT IS)
2046      /
2047      1211 3154      DCA GOBIT
2048      1212 4452      E70,   ERROR      /MISSING INTERRUPT
2049      1213 4502      T7OK,  SCOPE
2050      1214 5453      EXIT     / END OF TEST 7
2051      /TEST 10
2052      /
2053      /INTERRUPT TEST (PART III)
2054      /
2055      /IOT INTR 67X6 SHOULD CLEAR THE INTERRUPT ENABLE FLIP-FLOP, THEN
2056      /
2057      /WITH ALL FLAGS SET, NO INTERRUPTS SHOULD OCCUR
2058      /
2059      /TECHNICAL NOTE:
2060      /
2061      /IF AN UNEXPECTED PROGRAM INTERRUPT OCCURS FROM APPROXIMATELY THIS PC
2062      / THEN THE RX PROGRAM INTERRUPT REQUEST TOOK TOO LONG TO SET
2063      / FROM THE PREVIOUS TEST.
2064      /
2065      /* FOOTNOTE:
2066      /
2067      /THIS IOT "ION" IS ISSUED HERE BECAUSE - IF AN UNEXPECTED PROGRAM
2068      / INTERRUPT HAD OCCURED IN THE PREVIOUS TEST AND AC SW3 = 1 DIRECTING
2069      / THE PROGRAM NOT TO PRINT AN ERROR - THEN THE PDP'S INTERRUPT FACILITY
2070      / WOULD BE <OFF> - THEREFORE NEVER EXECUTING THIS TEST PROPERLY
2071      /
2072      1215 4507      T10,  WAITTY
2073      1216 4446      INTR     /DISABLE RX8 INTERRUPT ENABLE
2074      1217 1375      TAD (E100)
2075      1220 3002      DCA IPI

```

```

2076 1721 6001 ION / *SEE FOOTNOTE ABOVE
2077 1722 1377 TAD (200)
2078 1723 4436 LCD
2079 1724 7000 NOP
2080 1725 7410 SKP
2081 1726 4452 E100, ERROR /UNEXPECTED INTERRUPT
2082 1727 4502 SCOPE
2083 1730 5453 EXIT / END OF TEST 10
2084 /
2085 /
2086 /INTERRUPT TEST (PART IV)
2087 /
2088 /*****
2089 /*****
2090 /
2091 /IF AN RX01 CONTROL IS CABLED TO THE RX8 INTERFACE
2092 /THEN DON'T EXECUTE THIS TEST
2093 /BECAUSE THE CLEARING OF THE MAINTENANCE F/F
2094 /RESEMBLES A FILL BUFFER COMMAND
2095 /
2096 /CHECK TO SEE IF RUNNING ON A VT78 SYSTEM
2097 1731 4774 JMS VT78CK /VT78/ RETURN +1 IF NOT VT78
2098 1732 5236 JMP ,+4 /VT78 -ALWAYS EXECUTE TEST
2099 /
2100 1733 1165 T11, TAD RXHERE
2101 1734 7650 SMA CLA
2102 1735 5453 NOTEST
2103 /*****
2104 /*****
2105 /
2106 /TOGGLING THE MAINTENANCE MODE <ON> / <OFF> SETS ALL FLAGS AND
2107 /
2108 /PERMITS IOT SDN TO CLEAR THE DONE FLAG
2109 /
2110 /THEREFORE NO INTERRUPTS SHOULD OCCUR (ONLY DONE FLAG RAISES AN INTERRUPT REQUEST)
2111 /
2112 / (EVEN THOUGH THE RX01 INTERRUPT ENABLE IS 1 )
2113 /
2114 1736 4507 WAITTY
2115 /
2116 1737 1377 TAD (200)
2117 1740 4436 LCD
2118 1741 4436 LCD /MAINTENANCE <ON> / <OFF>
2119 1742 4445 SDN
2120 1743 7000 NOP
2121 /
2122 1744 1373 TAD (E110) /CLEAR THE DONE FLAG
2123 /
2124 1745 3002 DCA IPI
2125 1746 7201 CLA IAC
2126 1747 4446 INTR /RX01 INTERRUPT ENABLE <ON>
2127 /
2128 1751 7330 STL CLA RAR
2129 1752 4446 E110, INTR /RX01 INTERRUPT ENABLE <OFF>
2130 1753 1154 TAD GOBIT

```

```

2131 1754 7700 SMA CLA
2132 1755 4452 ERROR /UNEXPECTED INTERRUPT
2133 1756 4502 SCOPE
2134 1757 5453 EXIT / END OF TEST 11
2135 /TEST 12 - INITIALIZE TEST PART II (PROGRAMMED) / INTERRUPT TEST PART V
2136 /
2137 /*****
2138 /*****
2139 /
2140 /IF AN RX01 IS CABLED TO THE RX8 THEN DON'T EXECUTE T12
2141 /
2142 /BUT EXECUTE ALT12 (THE ALTERNATIVE TEST)
2143 /
2144 1760 1165 T12, TAD RXHERE
2145 1761 7650 SMA CLA
2146 1762 5772 JMP ALT12
2147 /*****
2148 /*****
2149 /
2150 /INTERRUPT TEST PART V / INITIALIZE TEST PART II (PROGRAMMED)
2151 /
2152 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT IOT INIT CLEARS THE INTERRUPT
2153 /ENABLE FLIP-FLOP WHEN SET
2154 /
2155 1763 1371 TAD (E124)
2156 1764 3002 DCA IPI
2157 1765 7201 CLA IAC
2158 1766 4446 INTR / SET THE RX8 INTERRUPT ENABLE F/F
2159 1767 4450 INITB
2160 1770 4451 CKUNIT /VT78/SETUP FOR UNIT A OR B
2161 1771 3154 DCA GOBIT / ISSUE INIT IOT 67X7
2162 /...BUT AN INTERRUPT SHOULD NOT OCCUR
2163 /
2164 /IF AN INTERRUPT OCCURS THEN IOT INIT FAILED TO CLEAR
2165 /
2166 /THE RX8 INTERRUPT ENABLE FLIP-FLOP
2167 /
2168 1772 1377 TAD (200)
2169 1773 4436 LCD
2170 1774 4436 LCD / MAINTENANCE MODE <ON> / <OFF>
2171 /
2172 /THE DONE FLAG SHOULD BE SET, BUT NO INTERRUPTS SHOULD OCCUR
2173 /
2174 1775 7330 STL CLA RAR
2175 /
2176 /RETURN TO HERE IF AN INTERRUPT OCCURED
2177 /
2178 1776 4446 E124, INTR / RX8 INTERRUPT ENABLE <OFF>
2179 1777 1154 TAD GOBIT
2180 1780 7700 SMA CLA
2181 1781 4452 ERROR / IOT INIT 67X7 DID NOT CLEAR THE IE F/F
2182 1782 4503 SUBSCOPE
2183 /
2184 /IOT "INIT" 67X7 SHOULD CLEAR THE RX8 INTERFACE TRANSFER REGISTER, THE
2185 /MAINTENANCE FLIP-FLOP, AND ALL FLAGS (DONE, TRANSFER REQUEST, AND ERROR).

```

```

2186 /
2187 1303 7240 STA
2188 1304 4436 LCD /ALL 1'S TO TRANSFER REGISTER
2189 1305 4450 INITB / IOT 67X7
2190 1306 4451 CKUNIT /VT78/SETUP FOR UNIT A OR B
2191 1307 4445 SDN
2192 1310 4465 OK
2193 1311 4452 E120, ERROR /UNEXPECTED DONE FLAG
2194 1312 4503 SUBSCOPE
2195 1313 4443 STR
2196 1314 4465 OK
2197 1315 4452 E121, ERROR /UNEXPECTED TRANSFER REQUEST FLAG
2198 1316 4503 SUBSCOPE
2199 1317 4444 SER
2200 1320 4465 OK
2201 1321 4452 E122, ERROR /UNEXPECTED ERROR FLAG
2202 1322 4503 SUBSCOPE
2203 1323 3166 DCA GOOD / PROGRAM EXPECTS TRANSFER REGISTER = 0
2204 1324 4441 XDRIN
2205 1325 7440 SZA
2206 1326 4452 E123, ERROR /TRANSFER REGISTER NOT = 0
2207 1327 4502 SCOPE
2208 /
2209 /*****
2210 /*****
2211 /
2212 /NO MORE RX8 INTERFACE TESTS EXIST
2213 /
2214 /IF AN RX01 CONTROL IS CABLED TO THE RX8 INTERFACE
2215 /
2216 /THEN CONTINUE WITH THE NORMAL FLOW OF TESTING
2217 /
2218 1100 XI=1100
2219 /
2220 / END OF PASS " I "
2221 /
2222 1330 4464 LOCKUP
2223 1331 1370 TAD (XI)
2224 1332 5767 JMP NOMORETESTS
2225 /*****
2226 /*****
2227 1333 1333 XSCOPE,
2228 1334 7300 CLA CLL
2229 1335 1365 TAD ERRORS
2230 1336 7640 SZA CLA
2231 1337 5357 JMP SCOPING
2232 /NO ERROR HAS BEEN DETECTED HERE

2233 /
2234 /JUST SET (PCSCOPE)=THE FIRST ADDRESS OF THE SCOPE LOOP
2235 /
2236 / (IN CASE ANY ERRORS ARE EVER DETECTED LATER)
2237 /
2238 1340 3365 NOSCOPE, DCA ERRORS
2239 1341 1333 TAD XSCOPE
2240 1342 3366 DCA PCSCOPE
    
```

```

2241 1343 5733 JMP I XSCOPE
2242 / " SUBSCOPE "
2243 /
2244 1344 1344 XSSCOPE,
2245 1345 1344 TAD XSSCOPE
2246 1346 3333 DCA XSCOPE
2247 1347 1365 TAD ERRORS
2248 1350 7650 SNA CLA
2249 1351 5733 JMP I XSCOPE
2250 /ERRORS DO EXIST
2251 /
2252 /IF THIS ERROR IS THE SAME AS THE ADDRESS WITHIN THE PROGRAM LOCATION
2253 /PCSSCOPE, THEN THIS IS A SCOPE LOOP
2254 /
2255 /IF NOT, THEN EXIT
2256 /
2257 1352 1333 TAD XSCOPE
2258 1353 7041 CIA
2259 1354 1364 TAD EPCSCOPE
2260 1355 7640 SZA CLA
2261 1356 5733 JMP I XSCOPE
2262 /THIS IS A SCOPING LOOP
2263 /
2264 1357 4572 SCOPING, LAS /TEST BIT 1
2265 1360 7004 RAL
2266 1361 7700 SNA CLA CLL
2267 1362 5340 JMP NOSCOPE
2268 1363 5766 JMP I PCSCOPE
2269 1364 0000 EPCSCOPE, 0 /ADDRESS +1 OF "SCOPE" OR "SUBSCOPE"
2270 1365 0000 ERRORS, 0 / > 0 IF AN ERROR HAS BEEN DETECTED (FOR THIS TEST)
2271 1366 0000 PCSCOPE, 0 / FIRST ADDRESS OF SCOPE LOOP
2272 1367 0465
2273 1370 1100
2274 1371 1276
2275 1372 1400
2276 1373 1252
2277 1374 1165
2278 1375 1226
2279 1376 1213
2280 1377 0200

2281 PAGE
2282 /ALTERNATE TEST 12 - VERIFICATION OF [INIT]
2283 /
2284 /INITIALIZE TEST PART II
2285 /
2286 /THIS TEST IS EXECUTED IN PLACE OF T12
2287 /
2288 /BECAUSE AN RX01 CONTROLLER IS CABLED TO THE RX8 INTERFACE
2289 /
2290 /IOT "INIT" 67X7 WILL PERFORM AN IMPLIED READ OF TRACK 0 SECTOR 1
2291 /
2292 / (IF DRIVE 0 IS READY)
2293 /
2294 /THEREFORE THE DONE FLAG SHOULD SET AT THE END OF THAT IMPLIED READ.
    
```

```

2295 /TECHNICAL NOTE:
2296 /
2297 /IF AN ERROR FLAG IS SET (AND DRIVE 0 IS READY) THEN THE ERROR MAY HAVE
2298 /BEEN THE RESULT FROM THE [IMPLIED READ SECTOR 0]
2299 /
2300 1400 4502 ALT12, SCOPE
2301 1401 4450 INITB
2302 1402 4451 CKUNIT /VT78/SETUP FOR UNIT A OR B
2303 1403 4443 ALT12LOOP, STR
2304 1404 7410 SKP
2305 1405 5212 JMP EA120 /UNEXPECTED TRANSFER REQUEST FLAG
2306 1406 4445 SDN
2307 1407 5203 JMP ALT12LOOP /WAIT FOR THE DONE FLAG
2308 1410 4443 STR
2309 1411 4465 OK
2310 1412 4452 EA120, ERROR /UNEXPECTED TRANSFER REQUEST
2311 1413 4503 SUBSCOPE
2312 1414 4444 SER
2313 1415 4465 OK
2314 1416 4452 EA121, ERROR /UNEXPECTED ERROR FLAG
2315 1417 4503 SUBSCOPE
2316 1420 1777 TAD UNITS /UNITS SELECTED BY OPERATOR
2317 1421 7710 SPA CLA
2318 1422 1376 TAD (40 /PROGRAM EXPECTS DRIVE 0 TO BE READY
2319 /
2320 /
2321 ///////////////////////////////////////////////////////////////////
2322 /
2323 / 4 5 - - 8 9 10 11 /
2324 /
2325 / SEL WRITE INIT PAR /
2326 / DRIVE DD PROTECT [DONE] CRC /
2327 / RDY (N/A) /
2328 /
2329 ///////////////////////////////////////////////////////////////////
2330 /
2331 /
2332 1423 7107 CLL IAC RFL / 4 [INIT] DONE, OR 204
2333 1424 3166 DCA GOOD
2334 1425 4441 XDRIN / "ACTUAL" STATUS AT DONE
2335 1426 3170 DCA BLANK
2336 1427 1170 TAD BLANK
2337 1430 0136 AND MASK /7677 FOR STANDARD 8 - 277 FOR VT78
2338 1431 3167 DCA EAC / "ACTUAL" MINUS DELETED DATA (IF ANY)
2339 1432 1167 TAD EAC
2340 1433 7041 CIA
2341 1434 1166 TAD GOOD /EXPECTED STATUS
2342 1435 7650 SNA CLA
2343 1436 5241 JMP ,+3 /COMPARED OK
2344 1437 1167 TAD EAC
2345 1440 4452 EA122, ERROR /TRANSFER REGISTER NOT = "GOOD"
2346 1441 4503 SUBSCOPE /NOTE;DRIVES NOT UNDER TEST MUST HAVE DOORS OPEN
2347 /
2348 /READ THE B-CODE STATUS AND EXPECT = 0
2349 /

```

```

2350 /PROGRAMMING NOTE:
2351 /
2352 /THE PROGRAM SAVES THE CONTENTS OF "GOOD" FOR REFERENCES WITHIN SUBROUTINE "XRST"
2353 /
2354 /BECAUSE "GOOD" IS REFRESHED WITHIN PROGRAM SUBROUTINE "XRSTB"
2355 /
2356 1442 1166 TAD GOOD
2357 1443 3170 DCA BLANK
2358 1444 4477 RSTB
2359 1445 7650 SNA CLA
2360 1446 5252 JMP ,+4
2361 1447 3166 DCA GOOD /PROGRAM EXPECTS 0
2362 1450 1167 TAD EAC
2363 1451 4452 EA123, ERROR /B-CODE NOT = 0
2364 1452 4503 SUBSCOPE
2365 /
2366 /READ THE CONTENTS OF THE RX01 STATUS REGISTER USING THE COMMAND # 5
2367 /
2368 /THIS STATUS SHOULD = THE STATUS IN THE TRANSFER REGISTER AT ERROR/DONE
2369 /
2370 1453 4476 RST / "READ STATUS" (COMMAND # 5)
2371 /
2372 1454 5453 EXIT / END OF TEST ALT12 (TEST 12)
2373 /THE PURPOSE OF THESE TESTS IS TO VERIFY THE TRANSFER LENGTH OF THE FUNCTION
2374 / "FILL BUFFER" AND "EMPTY BUFFER" OF THE RX01 MICROCONTROLLER
2375 /
2376 /64 TRANSFERS SHOULD OCCUR FOR 12-BIT MODE, AND
2377 /128 TRANSFERS SHOULD OCCUR FOR 8-BIT MODE
2378 /
2379 /THE SECTOR BUFFER IS FILLED WITH A COUNT PATTERN
2380 /
2381 / WORD/BYTE 0 = 0
2382 /
2383 / WORD 2 = 0202
2384 / BYTE 2 = 2
2385 /
2386 / WORD 77 = 7777
2387 / BYTE 177 = 177
2388 /
2389 /FILL BUFFER 8-BIT MODE
2390 /
2391 1455 1375 T16, TAD (100)
2392 /
2393 /FILL BUFFER 12-BIT MODE
2394 /
2395 1456 4436 T13, LCD /ISSUE THE COMMAND
2396 1457 1375 TAD (100)
2397 1460 0112 AND COMMAND
2398 1461 7640 SZA CLA
2399 1462 7307 CLL CLA IAC RFL / 4
2400 1463 1374 TAD (6000)
2401 1464 3132 DCA TESTP
2402 1465 4455 GETAPATTERN
2403 1466 1373 TAD (WBUFFER-1)
2404 1467 3010 DCA A10 /PROGRAMS "WRITE" BUFFER

```

```

2405 1470 1372 TAD (-WBUFFER+1)
2406 1471 3133 DCA XA10 / -STARTING ADDRESS OF WRITE BUFFER
2407 1472 4277 FB, JMS FBEB
2408 1473 5453 EXIT / ** END OF TESTS 13, OR 16
2409 1474 1410 TAD I A10
2410 1475 4442 XDROUT
2411 1476 5272 JMP FB / AND "FILL THE BUFFER"
2412 / FILL / EMPTY BUFFER SUBROUTINE FOR TESTS: *** 13, 16 / 14, 17 ***
2413 /
2414 1477 1477 FBEB, .
2415 1500 4443 STR
2416 1501 5304 JMP ,+3 /WAIT FOR TRANSFER REQUEST FLAG
2417 1502 2277 ISZ FBEB
2418 1503 5677 JMP I FBEB
2419 1504 4445 SDN
2420 1505 5300 JMP FBEB+1 /WAIT FOR THE DONE FLAG
2421 1506 3006 DCA COMMAND /FIRST TIME FOR WAIT,
2422 1507 1112 TAD COMMAND
2423 1510 3170 DCA BLANK
2424 1511 4444 SER
2425 1512 4485 OK
2426 1513 4452 E130, ERROR /UNEXPECTED ERROR FLAG
2427 1514 4503 SUBSCOPE
2428 /
2429 / 64 OR 128 BYTES SHOULD HAVE BEEN TRANSFERRED IN UR OUT
2430 /
2431 1515 1375 TAD (100)
2432 1516 0112 AND COMMAND / 200 FOR 8-BIT MODE
2433 1517 1375 TAD (100)
2434 1520 3166 DCA GOOD
2435 1521 1166 TAD GOOD
2436 1522 7041 CIA / 100 FOR 12-BIT MODE
2437 1523 1010 TAD A10
2438 1524 1133 TAD XA10
2439 1525 7440 SZA /SKIP IF TRANSFERS OK
2440 1526 4452 E131, ERROR / (AC) = - * MEANS NOT ENOUGH TRANSFERS
2441 1527 4502 SCOPE / (AC) > 0 MEANS TOO MANY TRANSFERS
2442 1530 5677 JMP I FBEB /RETURN IS TO EXIT
2443
2444
2445 /*****
2446 /CONSOLE PACKAGE
2447 /*****
2448 /*****
2449 /CONSUL
2450 /*****
2451

2452 1531 0000 PNTID, 0 /PRINT BEGIN MESSAGES
2453 1532 6007 CAF /TO REPLACE A CLEAR SWITCH
2454 1533 4470 APTH /TEST FOR APT SYSTEM,
2455 1534 4473 PRINT
2456 1535 6472 MIDENTIFICATION /ID MESSAGE
2457 1536 4473 PRINT
2458 1537 6453 REMOVE /REMOVE DIAGNOSTIC DISKETTE
2459 1540 4473 PRINT

```

```

2460 1541 6505 MSELECT /SELECT PARAMETERS
2461 1542 7200 CLA
2462 1543 3135 DCA UNITCK /CLEAR IT
2463 1544 5731 JMP I PNTID /EXIT PNTID
2464 /*****
2465 /
2466 /
2467 /
2468 /
2469 /
2470 /ROUTINE TO DETERMINE IF ON APT-8, IF APT-8 IS SELECTED
2471 /THEN CONSOLE AND TEST PARAMETER SELECTION FUNCTIONS ARE NOP,
2472 /IF NOT ROUTINE IS NOP,
2473 /
2474 1545 0000 XAPT8, 0
2475 1546 7300 CLA CLL
2476 1547 4472 CHEK22
2477 1550 7410 SKP /ON APT-8
2478 1551 5745 JMP I XAPT8
2479 1552 1022 TAD 22
2480 1553 0361 AND K7377 /NOP CONSOLE PACKAGE
2481 1554 3022 DCA 22 /RESTORE 22
2482 1555 1173 TAD K7000
2483 1556 3762 DCA I HLTNOP
2484 1557 1020 TAD 20 /GET TEST PARAMETERS,
2485 1560 5771 JMP AROUND-2 /MAIN FLOWOP PROGRAM,
2486 1561 7377 K7377, 7377
2487 1562 3316 HLTNOP, CBRET4-1
2488 1571 0231
2489 1572 0665
2490 1573 7113
2491 1574 6000
2492 1575 0100
2493 1576 0940
2494 1577 4235
2495 1600 PAGE
2496 /SECTOR BUFFER ADDRESSING VERIFICATION TESTS
2497 /
2498 /OPERATIONAL NOTE:
2499 /
2500 / (1). FOR TEST 14 TO EMPTY THE BUFFER IN 12-BIT MODE SUCCESSFULLY, TEST 13
2501 / MUST HAVE FILLED THE BUFFER IN 12-BIT MODE SUCCESSFULLY,
2502 /
2503 / (2). FOR TEST 17 TO EMPTY THE BUFFER IN 8-BIT MODE SUCCESSFULLY, TEST 16
2504 / MUST HAVE FILLED THE BUFFER IN 8-BIT MODE SUCCESSFULLY
2505 /
2506 /THE PURPOSE OF THESE TESTS IS TO VERIFY THAT THE CONTENTS OF THE SECTOR
2507 /BUFFER REMAIN UNCHANGED AFTER THE PREVIOUS EMPTY BUFFER 8-BIT MODE TEST, AND
2508 /AFTER THE PREVIOUS EMPTY BUFFER 12-BIT MODE TEST
2509 /
2509 1600 7410 T20, SKP /VERIFY EMPTY BUFFER 8-BIT MODE
2510 /
2511 1601 7410 T15, SKP /VERIFY EMPTY BUFFER 12-BIT MODE
2512 /
2513 /THE PURPOSE OF THESE TESTS IS TO VERIFY THE CONTENTS OF THE SECTOR BUFFER AFTER

```

```

2514 /THE PREVIOUS FILL BUFFER 8-BIT MODE TEST, AND THE PREVIOUS FILL BUFFER
2515 /12-BIT MODE TEST.
2516 /
2517 /EMPTY BUFFER 8-BIT MODE
2518 /
2519 1602 1377 T17, TAD (40
2520 /
2521 /EMPTY BUFFER 12-BIT MODE
2522 /
2523 1403 7105 T14, CLL IAC RAL
2524 1404 4436 LCD /ISSUE THE COMMAND 2 OR 102
2525 1405 1376 TAD (100)
2526 1406 0112 AND COMMAND
2527 1407 7640 SZA CLA
2528 1410 7307 CLL CLA IAC RTL
2529 1411 1375 TAD (6000)
2530 1412 3132 DCA TESTP
2531 1413 4455 GETAPATTERN / COUNT PATTERN ( PATTERN #6)
2532 1414 1374 TAD (RBUFFER-1)
2533 1415 3010 DCA A10 /PROGRAMS "READ" BUFFER
2534 1416 1373 TAD (-RBUFFER+1)
2535 1417 3133 DCA XA10 / -STARTING ADDRESS OF READ BUFFER
2536 1420 1010 TAD A10
2537 1421 3011 DCA A11 /
2538 1422 4772 EB, JMS FBEB
2539 1423 5230 JMP EBCOMPARE
2540 1424 3410 DCA I A10
2541 1425 4441 XDRIN
2542 1426 3411 DCA I A11 / AND "EMPTY THE BUFFER"
2543 1427 5222 JMP EB /
2544 /COMPARE THE CONTENTS OF THE SECTOR BUFFER
2545 /
2546 /WITH THE GOOD DATA IN "WBUFFER"
2547 /
2548 1430 1371 ERCOMPARE, TAD (WBUFFER-1)
2549 1431 3133 DCA XA10 /EXPECTED
2550 1432 1374 TAD (RBUFFER-1)
2551 1433 3134 DCA XA11 /ACTUAL
2552 1434 3113 DCA COMPERROROR / = 1 IF COMPARE ERROR
2553 /
2554 1435 2133 EBLOOP, ISZ XA10
2555 1436 2134 ISZ XA11
2556 1437 7100 CLL
2557 1440 1376 IAD (100)
2558 1441 0112 AND COMMAND
2559 1442 7640 SZA CLA
2560 1443 7120 STL /SET LINK IF 8-BIT MODE

2561 1444 1533 TAD I XA10
2562 1445 7430 SZL
2563 1446 0370 AND (377)
2564 1447 3166 DCA GOOD
2565 1450 1166 TAD GOOD
2566 1451 7041 CIA
2567 1452 1534 TAD I XA11
2568 1453 7650 SNA CLA
    
```

```

2569 1454 5263 JMP EBUX
2570 /A COMPARE ERROR HAS OCCURED
2571 /
2572 /INCORRECT DATA WAS TRANSFERRED FROM THE RX01 CONTROL SECTOR BUFFER
2573 /TO THE RX8 INTERFACE AND SAVED WITHIN PROGRAM LOCATIONS BEGINNING WITH "RBUFFER"
2574 /
2575 /THAT DATA, HOWEVER, MAY HAVE BEEN TRANSFERRED INCORRECTLY *TO*
2576 /THE RX01 CONTROL FROM THE RX8 INTERFACE PREVIOUSLY WITHIN T12, OR T15
2577 /
2578 1455 1367 TAD (-WBUFFER)
2579 1456 1133 TAD XA10
2580 1457 3170 DCA BLANK
2581 1460 1534 TAD I XA11
2582 1461 4452 E140, ERROR /ACTUAL WORD/BYTE FROM SECTOR BUFFER
2583 1462 4502 SCOPE /IS NOT = EXPECTED)
2584 1463 1376 EROK, TAD (100)
2585 1464 0112 AND COMMAND
2586 1465 1376 IAD (100)
2587 1466 7041 CIA
2588 1467 1133 TAD XA10
2589 1470 1366 TAD (-WBUFFER+1)
2590 1471 7640 SZA CLA
2591 1472 5235 JMP EBLOOP
2592 /END OF TESTS 14, 15, 17, OR 20
2593 /
2594 /...ALSO
2595 /
2596 /END OF TESTS 31, 32, OR 33
2597 /
2598 1473 5453 EXIT
2599 /
2600 /SECTOR BUFFER DATA TESTING
2601 /
2602 /THE PURPOSE OF THESE TESTS IS TO VERIFY THAT ALL 1'S AND ALL 0'S CAN BE
2603 /SET INTO THE SECTOR BUFFER
2604 /
2605 /SECTOR BUFFER DATA TESTING
2606 /
2607 /FILL THE SECTOR BUFFER WITH ALL 1'S
2608 /
2609 1474 1370 T22, TAD (377) / "GOOD " = ALL 1'S
2610 /
2611 /FILL THE SECTOR BUFFER WITH ALL 0'S
2612 /
2613 /FILL THE SECTOR BUFFER WITH 128 BYTES OF "GOOD"
2614 /
2615 1475 4765 T21, JMS FB128BYTES
2616 1476 4764 JMS TX / "GOOD " = ALL 0'S
2617 /*****
2618 /*****
2619 /
2620 /IF THIS IS TEST #22
2621 /
2622 /THEN TEST FOR A DRIVE SELECTION
2623 /
    
```

```

2624 /IF NO DRIVES ARE ENABLED WITHIN PROGRAM LOCATION " DTESTP "
2625 /
2626 /THEN THERE ARE NO MORE RX01 CONTROL TESTS TO EXECUTE
2627 /
2628 TAD TEST
2629 TAD (-T22)
2630 SZA CLA
2631 EXIT / END OF TEST 21
2632 TAD UNITS
2633 SZA CLA
2634 EXIT / END OF TEST 22
2635 XC=0300
2636 /
2637 / END OF PASS " C "
2638 /
2639 CLL CLA CMA
2640 DCA COUNT /INIT TIMING FOR APT IF ONLY
2641 /INTERFACE IS TO BE TESTED.
2642 JMS XTICK
2643 LOCKUP
2644 TAD (XC)
2645 JMP NOMORETESTS
2646 /*****
2647 /*****
2648 T24, CLA STL IAC RTL
2649 JMP IRDWR / 6 (READ)
2650 T25, CLL CLA IAC RTL
2651 JMP IRDWR / 4 (WRITE)
2652 T26, TAD (14)
2653 JMP IRDWR / 14 (WRITE DELETED DATA)
2654 /
2655 /THE PURPOSE OF THESE TESTS IS TO VERIFY
2656 /THE RX01 CONTROL CLOCK SET OF THE ERROR FLAG
2657 /
2658 /BY FORCING A SEEK ERROR TO OCCUR
2659 / (ATTEMPTING TO PERFORM A FUNCTION ON A NON-EXISTANT SECTOR #0 )
2660 /
2661 /NOTE:THE CONTENTS OF THE SECTOR BUFFER SHOULD REMAIN UNCHANGED AND CONTAIN
2662 / THE PATTERN OF ALL 1'S AS FILLED WITHIN TEST 22 BECAUSE THE READ SHOULD
2663 / HAVE NEVER OCCURED.
2664 /
2665 / 1. THE STATUS WITHIN THE TRANSFER REGISTER AT ERROR SHOULD = 0
2666 / 2. THE B-CODE STATUS SHOULD = 70 (UNABLE TO FIND SECTOR)
2667 / 3. THE STATUS FROM THE RST COMMAND (12) SHOULD = DRIVE READY (200)
2668 /
2669 /THE LENGTH ( # OF TRANSFERS TO THE RX01 CONTROL) SHOULD BE TWO
2670 / (ONE EACH FOR THE SECTOR AND FOR THE TRACK)
2671 /
2672 /GET ONE UNIT ONLY (UNJT 1 IF UNIT 0 IS NOT READY)
2673 /
2674 /ROUTINE TO WAIT FOR SKIP ON AN IOT, IF SKIP DOES NOT OCCUR
2675 /THE ROUTINE WILL PRINT PC POINT IN ERROR AND GO BACK ABOUT ITS
2676 /BUSINESS.
2677 /
2678 1722 0000 XWAIT, 0

```

```

2679 1723 1006 TAD 6
2680 1724 7650 SNA CLA
2681 1725 5330 JMP ,+3
2682 1726 7240 STA
2683 1727 3006 DCA 6
2684 1730 2153 ISZ H1
2685 1731 5722 JMP I XWAIT
2686 1732 2116 ISZ HANGER
2687 1733 5722 JMP I XWAIT
2688 1734 1322 TAD XWAIT /GET ERROR PC
2689 1735 1354 TAD (=E3PRE
2690 1736 7710 SPA CLA /DID CALL COME FROM PRETEST.
2691 1737 5753 JMP E3PRE /YES, REPRORT ERROR.
2692 1740 1322 TAD XWAIT /GET BACK ERROR
2693 1741 5752 JMP HUNGUP
2694 /
2695 1752 3341
2696 1753 0425
2697 1754 7353
2698 1755 0014
2699 1756 2000
2700 1757 0300
2701 1760 4133
2702 1761 4152
2703 1762 4235
2704 1763 6104
2705 1764 2303
2706 1765 4541
2707 1766 0665
2708 1767 0664
2709 1770 0377
2710 1771 7113
2711 1772 1477
2712 1773 0465
2713 1774 7313
2714 1775 6000
2715 1776 0100
2716 1777 0040
2717 2000 3112 PAGE
2718 2001 3777 IRDWR, DCA COMMAND
2719 2002 4460 DCA WUNITS
2720 2003 1112 GETUNIT
2721 2004 1776 TAD COMMAND
2722 2005 3170 TAD UNIT
2723 2006 1170 DCA BLANK
2724 2007 4436 TAD BLANK / 6, 4, OR 14
2725 2010 3167 LCD
2726 2011 5214 DCA EAC
2727 2012 2167 JMP ,+3
2728 2013 4442 ISZ EAC
2729 2014 4443 XDROUT
2730 2015 7410 STR
2731 2016 5212 SKP
2732 2017 4445 JMP ,=4 /SECTOR 0 - TRACK 0
SDN

```



```

2733 2020 5214 JMP ,=-4 /WAIT FOR DONE FLAG
2734 2021 7344 CLL STA RAL
2735 2022 1167 TAD EAC
2736 2023 7650 SNA CLA
2737 2024 5231 JMP ,+5
2738 2025 7305 CLL CLA IAC RAL
2739 2026 3166 DCA GOOD / 2 TRANSFERS WERE EXPECTED
2740 2027 1167 TAD EAC
2741 2030 4452 E240, ERROR / # OF TRANSFER REQUEST FLAGS NOT OK
2742 2031 4503 SUBSCOPE / (AC) = # OF TRANSFERS OCCURED
2743 /
2744 /THE ERROR FLAG SHOULD = 1
2745 /
2746 2032 4444 SER
2747 2033 4452 E245, ERROR / MISSING ERROR FLAG
2748 2034 4503 SUBSCOPE
2749 /IF THIS IS T24, THEN DELETED DATA (MAY) BE SET (BUT THAT'S OK FOR NOW),
2750 /THE STATUS AT ERROR SHOULD = X (100 MAYBE DELETED DATA)+200 DRIVE READY
2751 /
2752 /IF THIS IS T25, THEN DELETED DATA (SHOULD NOT) BE SET, THEREFORE
2753 /THE CONTENTS OF THE TRANSFER REGISTER (THE STATUS AT THE ERROR) SHOULD = 200
2754 /
2755 /IF THIS IS T26, THEN DELETED DATA (MUST) BE SET, THEREFORE
2756 /THE STATUS SHOULD = 300 (200 DRIVE READY)+(100 (DELETED DATA)
2757 /
2758 2035 1171 TAD TEST /FOR T #
2759 2036 1375 TAD (-T25)
2760 2037 7650 SNA CLA
2761 2040 5245 JMP ,+5 / T25 THEN " JMP ,+5 "
2762 2041 1171 TAD TEST
2763 2042 1374 TAD (-T26)
2764 2043 7650 SNA CLA
2765 2044 1373 TAD (100) / T26 MEANS EXPECT DELETED DATA
2766 2045 3166 DCA GOOD / T24
2767 2046 4441 XORIN /ACTUAL STATUS
2768 2047 3170 DCA BLANK
2769 2050 1170 TAD BLANK
2770 2051 0372 AND (377) /VT78/
2771 2052 3167 DCA EAC
2772 /
2773 /IF THIS IS T24, THEN DELETED DATA (MAY) BE SET, (BUT THAT'S OK FOR NOW)
2774 /
2775 2053 1171 TAD TEST / FOR T #
2776 2054 1371 TAD (-T24)
2777 2055 7640 SZA CLA / T24 ?
2778 2056 5202 JMP ,+4
2779 2057 1136 TAD MASK / YES
2780 2060 0170 AND BLANK /ACTUAL STATUS MINUS DELETED DATA
2781 2061 3167 DCA EAC
2782 2062 1167 TAD EAC
2783 2063 7041 CIA
2784 2064 1166 TAD GOOD /EXPECTED
2785 2065 7650 SNA CLA
2786 2066 5271 JMP ,+3
2787 2067 1167 TAD EAC

```

```

2788 2070 4452 E241, ERROR /STATUS NOT = "GOOD"
2789 2071 4503 SUBSCOPE
2790 /
2791 /THE B-CODE SHOULD = 70 (UNABLE TO FIND SECTOR)
2792 /
2793 2072 4477 RSTB /RETURN WITH AC = CODE
2794 2073 1370 TAD (-70)
2795 2074 7650 SNA CLA
2796 2075 5302 JMP ,+5
2797 2076 1367 TAD (70)
2798 2077 3166 DCA GOOD
2799 2100 1167 TAD EAC
2800 2101 4452 E242, ERROR /B-CODE STATUS NOT = CODE # 70
2801 2102 4502 SCOPE
2802 /THE CONTENTS TO THE SECTOR BUFFER SHOULD REMAIN UNCHANGED
2803 /
2804 /THE CONTENTS OF THE SECTOR BUFFER SHOULD = ALL BYTES OF 1'S
2805 /
2806 2103 1372 TAD (377)
2807 2104 3166 DCA GOOD /EXPECT ALL 1'S
2808 2105 4766 JMS TX /VERIFY SECTOR BUFFER SUBROUTINE
2809 2106 5453 EXIT / END OF TEST 24, 25, 26
2810 /
2811 / TEST 27 - SEEK AND CRC VERIFICATION (FIRST PROGRAMMED HEAD MOVEMENT)
2812 /
2813 /THE PURPOSE OF THIS TEST IS TO VERIFY THAT SUPPLYING THE RX01 WITH A TRACK
2814 /WHOSE VALUE IS GREATER THAN 114 (OCTAL) EXPECTS A B-CODE ERROR OF 40
2815 /
2816 2107 1365 T27, TAD (115)
2817 2110 3170 DCA BLANK
2818 2111 7327 STL CLA IAC RTL / ISSUE READ COMMAND
2819 2112 4436 LCD
2820 2113 3167 DCA EAC
2821 2114 7410 SKP
2822 2115 2167 ISZ EAC / + TO TRANSFER COUNT
2823 2116 1170 TAD BLANK / SECTOR TRACK
2824 2117 4442 XDRGUT /TO RX01 CONTROL
2825 2120 4443 STR / WAIT FOR TRANSFER REQUEST FLAG
2826 2121 7410 SKP
2827 2122 5315 JMP ,=-5
2828 2123 4445 SDN / WAIT FOR DONE FLAG
2829 2124 5320 JMP ,=-4
2830 2125 3006 DCA 6 /WAIT POINTER
2831 2126 7344 CLL STA RAL / ONLY 2 TRANSFER REQUESTS WERE EXPECTED
2832 2127 1167 TAD EAC / ACTUAL # OF TRANSFER REQUEST OCCURED
2833 2130 7650 SNA CLA
2834 2131 5336 JMP ,+5
2835 2132 7305 CLL CLA IAC RAL
2836 2133 3166 DCA GOOD
2837 2134 1167 TAD EAC
2838 2135 4452 E270, ERROR / # OF TRANSFER REQUEST FLAGS NOT OK
2839 2136 4503 SUBSCOPE / (AC) CONTAINS THE # OF REQUEST OCCURED
2840 /
2841 /THE ERROR FLAG SHOULD = 1
2842 /

```

```

2843 2137 4444 SER
2844 2140 4452 E271, ERROR / MISSING ERROR FLAG
2845 2141 4503 SUBSCOPE
2846 /THE B-CODE SHOULD = 40
2847 /
2848 2142 4477 RSTB
2849 2143 1364 TAD (-40)
2850 2144 7650 SNA CLA
2851 2145 5352 JMP ,+5
2852 2146 1363 TAD (40)
2853 2147 3166 DCA GOOD
2854 2150 1167 TAD EAC
2855 2151 4452 E272, ERROR / B-CODE NOT = 40
2856 2152 4502 SCOPE
2857 /
2858 /THE CONTENTS OF THE SECTOR BUFFER SHOULD REMAIN UNCHANGED
2859 /
2860 2153 1372 TAD (377)
2861 2154 3166 DCA GOOD / EXPECT ALL 1'S
2862 2155 4766 JMS TX
2863 /
2864 2156 5453 EXIT / END OF TEST 27
2865 2163 0040
2866 2164 7740
2867 2165 0115
2868 2166 2303
2869 2167 0070
2870 2170 7710
2871 2171 6064
2872 2172 0377
2873 2173 0100
2874 2174 6060
2875 2175 6062
2876 2176 4242
2877 2177 4236
2878 2200 PAGE
2879 /TEST 30 - SEEK AND CRC VERIFICATION
2880 /
2881 /READ ALL SECTORS OF ALL TRACKS
2882 /
2883 /IF THE DATA IS OF KNOWN QUALITY THEN MONITOR FOR CRC EPRORS
2884 2700 3132 T30, DCA TESTP
2885 2701 5777 JMP TEST4
2886 /TEST 31 - FIRST WRITE EVER
2887 /
2888 /WRITING TO THE DISK SHOULD NOT DESTROY THE CONTENTS OF THE SECTOR BUFFER
2889 /
2890 /TECHNICAL NOTE:
2891 /
2892 /THIS TEST WRITES ON ONE UNIT ONLY - UNIT 0, BUT IF UNIT 0 WAS
2893 /NOT SELECTED BY THE OPERATOR AT THE START OF THIS PROGRAM
2894 /THEN THIS TEST WRITES ON UNIT 1
2895 /
2896 /PROGRAMMING NOTE:

```

```

2897 /
2898 /THIS "DCA WUNITS" IS NECESSARY IF THIS TEST IS LOCKED WITH SWS
2899 /
2900 2702 3776 T31, DCA WUNITS
2901 /
2902 2703 7307 CLL CLA IAC RTL / 4 (WRITE 8-BIT MODE)
2903 2204 1375 TAD (6000)
2904 2705 3132 DCA TESTP /COUNT PATTERN 6
2905 2706 7240 STA
2906 2707 3120 DCA RDC
2907 / "GETATRACK"
2908 /
2909 2710 1027 TAD OD
2910 2711 3131 DCA TARGET / TRACK = (OD)
2911 / "GETASECTOR"
2912 /
2913 2712 1031 TAD FIRST
2914 2713 3124 UCA STARGET / SECTOR = (FIRST)
2915 /GET UNIT 0 (OR UNIT 1 IF UNIT 0 NOT SELECTED)
2916 /
2917 2714 4460 GETUNIT
2918 /
2919 / "INITSECTOR"
2920 /
2921 2215 7240 STA
2922 2216 3122 DCA SECTORS / 1 SECTOR (FIRST) TO WRITE
2923 2217 1374 TAD (+3)
2924 2220 3773 DCA XWRITE / RETURN ADDRESS FROM WRITE SUBROUTINE
2925 2221 5772 JMP REWRITE+1 / JMP TO WRITE SUBROUTINE
2926 /
2927 /RETURN HERE FROM SUBROUTINE " XWRITE "
2928 /
2929 / JMP TO T17 TO VERIFY THE CONTENTS OF THE SECTOR BUFFER
2930 /
2931 2222 5771 JMP T17
2932 /TEST 32 - INIT [PROGRAMMED] PART III / IMPLIED READ OF TRACK 1 SECTOR 1
2933 /
2934 /*****
2935 /*****
2936 /
2937 /IF THE CONTENTS OF PROGRAM LOCATIONS:
2938 /
2939 / " OD " = 1, AND
2940 / " FIRST " = 1,
2941 /
2942 /AND IF UNIT 0 WAS SELECTED BY THE OPERATOR AT THE START OF THIS PROGRAM
2943 /
2944 /THEN EXECUTE THIS TEST
2945 /
2946 2223 1031 T32, TAD FIRST
2947 2224 7110 CLL RAR
2948 2225 7640 SZA CLA
2949 2226 5453 NOTEST
2950 2227 1027 TAD OD
2951 2230 7110 CLL RAR

```

```

2952 2731 7640          SZA CLA
2953 2732 5453          NOTEST
2954 2733 1770*        TAD UNITS
2955 2734 7700          SMA CLA
2956 2735 5453          NOTEST
2957
2958 /*****
2959 /*****
2960 /
2961 /THE PURPOSE OF THIS TEST IS TO VERIFY THE "IMPLIED READ" OF TRACK 1 SECTOR 1
2962 /AS PART OF THE INITIALIZE FUNCTION
2963 /
2964 /FIRST FILL THE SECTOR BUFFER WITH ALL 0'S, THEN ISSUE IOT INIT 67X7
2965 /
2966 /INIT SHOULD READ THE CONTENTS OF TRACK 1 SECTOR 1 OF UNIT 0
2967 /
2968 /INTO THE SECTOR BUFFER
2969 /
2970 2736 4767*          JMS FB128BYTES          / FILL THE SECTOR BUFFER
2971 2737 4447          INIT                      /OF UNIT 0
2972 /
2973 /JMP TO TEST 17 TO VERIFY THE CONTENTS OF THE SECTOR BUFFER
2974 /
2975 2740 5771*          JMP T17                      / TO TEST 17
2976 /TEST 33 - FIRST READ [PROGRAMMED] EVER
2977 /
2978 /FIRST FILL THE SECTOR BUFFER WITH ALL 0'S
2979 /
2980 /THEN READ FROM THE DISK TRACK # (OD), SECTOR # (FIRST)
2981 /
2982 /THE CONTENTS OF THE SECTOR BUFFER SHOULD BE THAT OF THE PREVIOUS TEST
2983 /
2984 /PROGRAMMING NOTE:
2985 /
2986 /THIS "DCA WUNITS" IMPERATIVE IF T27 WAS EXECUTED PREVIOUSLY THIS PASS
2987 /
2988 2741 3776*          T33,   DCA WUNITS
2989 2742 4767*          JMS FB128BYTES          / FILL THE SECTOR BUFFER
2990 2743 7307          CLL CLA IAC RTL
2991 2744 3132          DCA TESTP          / 4 (READ 8-BIT MODE)
2992 /PROGRAMMING NOTE:
2993 /
2994 /THE FOLLOWING CODE TO " JMP T17 " IS IMPERATIVE HOUSEKEEPING PRIMING THE
2995 / " READ " SUBROUTINE
2996 /
2997 2745 7240          STA
2998 2746 3120          DCA RDC

2999 / "GETATRACK"
3000 /
3001 2747 1027          TAD OD
3002 2750 3131          DCA TARGET
3003 / "GETASECTOR"
3004 /
3005 2751 1031          TAD FIRST
3006 2752 3124          DCA STARGET
    
```

```

3007 /GET ONE UNIT ONLY (UNIT 1 IF UNIT 0 IS NOT READY)
3008 /
3009 2753 4460          GETUNIT          / SELECT A UNIT
3010 2754 7240          STA
3011 2755 3140          DCA RIRETRY          /SO NO "REWRITES" OCCUR
3012 / "INITSECTOR"
3013 /
3014 2756 7240          STA
3015 2757 3122          DCA SECTORS          / 1 SECTOR TO READ (#FIRST)
3016 2760 1366          TAD (.,+4)
3017 2761 3765*        DCA XREAD          /RETURN ADDRESS FROM " READ " SUBROUTINE
3018 2762 5764*        JMP READRETRY          / JMP TO READ SUBROUTINE
3019 /THIS "WRITE" IS IMPERATIVE FOR REFERENCES WITHIN "XREAD"
3020 /
3021 2763 4510          WRITE          / "WRITE" FOR PROGRAM REFERENCES ONLY
3022 /
3023 /RETURN HERE FROM SUBROUTINE " XREAD "
3024 /
3025 /
3026 /JMP TO TEST 17 TO VERIFY THE CONTENTS OF THE SECTOR BUFFER
3027 /
3028 2764 5771*          JMP T17                      / TO TEST 17
3029 /TEST 23 - DRIVE READY SELECTION (ALL UNITS SELECTED BY OPERATOR)
3030 /
3031 /THE PURPOSE OF THIS TEST IS TO VERIFY THE "SEL DRV RDY" STATUS BIT 4
3032 /
3033 2765 1363          T23,   TAD (200)
3034 2766 3170          DCA BLANK
3035 2767 4460          GETUNIT
3036 2770 1762*        TAD UNIT
3037 2771 4476          RST
3038 2772 4454          DONE
3039 2773 5265          JMP T23
3040 2774 5453          EXIT                      / END OF TEST 27
3041 /
3042 /THE PURPOSE OF THESE TESTS IS TO WRITE-READ-AND PROGRAM VERIFY THE DATA
3043 /ON ALL TRACKS FROM (OD) TO (ID), AND ALL SECTORS FROM (FIRST) TO (LAST),
3044 /EXERCISING 8-BIT MODE, 12-BIT MODE, AND DELETED DATA
3045 /
3046 /A PATTERN OF ALL 1'S ARE WRITTEN ON THE DISK
3047 /
3048 /
3049 /WORD/BYTE 1 IS THE TRACK ID (BITS 5 THRU 11)
3050 /WORD/BYTE 2 IS THE SECTOR ID (BITS 7 THRU 11)
3051 /
3052 /
3053 /WORDS 3 THRU 62 IS THE ALL 1'S DATA
3054 /
3055 /
3056 /BYTES 3 THRU 126 IS THE ALL 1'S DATA
3057 /
3058 /
3059 /WORDS 63, AND 64 ARE SPECIAL SUMCHECK WORDS
3060 /
3061 /
3062 /BYTES 127, AND 128 ARE SPECIAL SUMCHECK WORDS
3063 /
3064 /TEST 37 - 8 BIT MODE WITH DELETED DATA
3065 /
3066 2775 7305          T37,   CLL CLA IAC RAL          / 6 (DELETED DATA = 8/BIT MODE)
    
```

```

3062 /
3063 /TEST 35 = 8 BIT MODE
3064 /
3065 2276 1361 T35, TAD (2) / 4 (8/BIT MODE)
3066 /
3067 /TEST 36 = 12 BIT MODE WITH DELETED DATA
3068 /
3069 2277 1361 T36, TAD (2) / 2 (DELETED DATA = 12/BIT MODE)
3070 /
3071 /TEST 34 = 12 BIT MODE
3072 /
3073 2300 1360 T34, TAD (1000) /ALL 1'S PATTERN
3074 2301 3132 DCA TESTP / (12/BIT MODE)
3075 2302 5757 JMP THETEST
3076 /THIS SUBROUTINE IS ENTERED FROM TESTS: *** T21, T22, (T24, T25, T26), T27
3077 /
3078 /EMPTY THE BUFFER TO VERIFY THE CONTENTS = ALL 1'S OR ALL 0'S
3079 /
3080 2303 2303 TX, .
3081 2304 3170 DCA BLANK / BYTE # 1 TO 128
3082 2305 1356 TAD (102)
3083 2306 4436 LCD /EMPTY BUFFER 8-BIT MODE
3084 2307 5322 JMP T20STR=1
3085 2310 4441 T20XDRIN, XDRIN
3086 2311 3167 DCA EAC /DATA FROM SECTOR BUFFER
3087 2312 1167 TAD EAC
3088 2313 7041 CIA
3089 2314 1166 TAD GOOD /COMPARED WITH EXPECTED DATA
3090 2315 7650 SNA CLA
3091 2316 5321 JMP ,+3
3092 2317 1167 TAD EAC
3093 2320 4452 E211, ERROR /DATA "TO" NOT = DATA "FROM"
3094 2321 4503 SUBSCOPE
3095 2322 2170 ISZ BLANK / 1 TO 128
3096 2323 4443 T20STR, STR
3097 2324 7410 SKP
3098 2325 5310 JMP T20XDRIN
3099 2326 4445 SDN
3100 2327 5323 JMP T20STR
3101 2330 4444 SER
3102 2331 4465 OK
3103 2332 4452 E212, ERROR /UNEXPECTED ERROR FLAG
3104 2333 4502 SCOPE
3105 2334 5703 JMP I TX
3106 /VT78/ POUTINE TO RESELECT UNIT UNDER TEST.
3107 /ON VT78 A CAF OR INIT INSTRUCTION AUTOMATICALLY
3108 /RESELECTS UNIT A(RXA).

3109 /THE FLAGS ARE CLEARED AT THIS TIME DUE TO POSSIBLE
3110 /GLITCHES IN THE RX8 INTERFACE.
3111 2335 0000 XCKUNT, 0
3112 2336 7300 CLA CLL
3113 2337 1135 TAD UNITCK /0=UNIT A 1=UNIT B
3114 2340 4435 SEL /RESETS UNIT SELECT TO UNIT UNDER TEST
3115 2341 4445 SDN /CLEAR FLAGS
3116 2342 7000 NOP
    
```

```

3117 2343 4444 SER
3118 2344 7000 NOP
3119 2345 4443 STR
3120 2346 7300 CLA CLL
3121 2347 5735 JMP I XCKUNT
3122 2356 0102
3123 2357 3000
3124 2360 1000
3125 2361 0002
3126 2362 4242
3127 2363 0200
3128 2364 3425
3129 2365 3416
3130 2366 2264
3131 2367 4541
3132 2370 4235
3133 2371 1602
3134 2372 3207
3135 2373 3200
3136 2374 2222
3137 2375 6000
3138 2376 4236
3139 2377 2514
3140 2400 PAGE
3141 /READ THE B-CODE STATUS
3142 /
3143 /THIS SUBROUTINE IS ENTERED FROM TESTS: *** ALT12, (T24, T25, T26), T27
3144 2400 2400 XRSTB, .
3145 2401 1377 TAD (16)
3146 2402 4436 LCD /ISSUE COMMAND # 7
3147 2403 4443 STR
3148 2404 4465 OK
3149 2405 4452 E7000, ERROR /UNEXPECTED TRANSFER REQUEST FLAG
3150 2406 4503 SUBSCOPE
3151 2407 4445 SDN
3152 2410 5203 JMP ,=5
3153 2411 4441 XDRIN
3154 2412 3167 DCA EAC /ACTUAL STATUS
3155 2413 4444 SER
3156 2414 5217 JMP ,+3
3157 2415 1167 TAD EAC
3158 2416 4452 E7001, ERROR /UNEXPECTED ERROR FLAG
3159 2417 4776 JMS VT78CK /GO CHECK IF ON VT78 SYSTEM
3160 2420 5233 JMP E7002+1 /VT78 = SKIP THIS SUBTEST
3161 2421 4503 SUBSCOPE
3162 2422 1167 TAD EAC
3163 2423 0375 AND (7000) /BITS 0,1,2 ARE RESIDUAL FROM COMMAND SHIFT
3164 2424 1374 TAD (1000)
3165 2425 7650 SNA CLA
3166 2426 5233 JMP ,+5
3167 2427 1375 TAD (7000)
3168 2430 3166 DCA GOOD
3169 2431 1167 TAD EAC
3170 2432 4452 E7002, ERROR / SHIFT REGISTER NOT SHIFTING OK
    
```

```

3171 2433 4503          SUBSCOPE
3172 2434 1167          TAD EAC
3173 2435 0373          AND (377)
3174 2436 3167          DCA EAC
3175 2437 1167          TAD EAC
3176 2440 5600          JMP I XRSTB
3177
3178          /READ STATUS SUBROUTINE
3179
3180          /THE CONTENTS OF THE AC AT ENTRY = BIT 7 (UNIT SELECT)
3181
3182          /READ THE STATUS AT DONE BY ISSUING COMMAND # 5
3183
3184          /THIS SUBROUTINE IS ENTERED FROM TESTS: *** ALT12, AND T27
3185
3186 2441 2441          XRST,
3187 2442 1372          TAD (12)
3188 2443 4436          LCD          / ISSUE COMMAND # 5 (AC AT ENTRY = UNIT)
3189 2444 4443          STR
3190 2445 4465          OK
3191 2446 4452          E7003, ERROR          /UNEXPECTED TRANSFER REQUEST FLAG
3192 2447 4503          SUBSCOPE
3193 2450 4445          SDN
3194 2451 5244          JMP ,=5          /WAIT FOR DONE FLAG
3195 2452 4441          XDRIN
3196 2453 3167          DCA EAC          /ACTUAL STATUS
3197 2454 4444          SER
3198 2455 5260          JMP ,+3
3199 2456 1167          TAD EAC
3200 2457 4452          E7004, ERROR          /UNEXPECTED ERROR FLAG
3201 2460 4776          JMS VT78CK          /GO CHECK IF ON VT78 SYSTEM
3202 2461 5274          JMP E7006+1          /VT78 = SKIP THIS SUBTEST
3203 2162 4503          SUBSCOPE
3204
3205          /THE PURPOSE OF THIS SUBTEST IS TO VERIFY THE SHIFTING OF THE RX8 INTERFACE
3206
3207          /TRANSFER REGISTER BY THE RX01 MICROCONTROLLET
3208
3209          /BITS 0, 1, AND 2 OF THE RX8 INTERFACE TRANSFER REGISTER SHOULD BE REMNANTS
3210
3211          /OF THE PREVIOUS COMMAND (COMMAND # 5), THEREFORE BITS 0, 1, AND 2 SHOULD = 101 (BINARY)
3212          /{(NOT TRUE ON VT78)}
3213 2463 1167          TAD EAC
3214 2464 0375          AND (7000)
3215 2465 1371          IAD (3000)
3216 2466 7650          SNA CLA
3217 2467 5274          JMP ,+5
3218 2470 1370          TAD (5000)
3219 2471 3166          DCA GOOD
3220 2472 1167          TAD EAC
3221 2473 4452          E7006, EPRON
3222 2474 4503          SUBSCOPE
3223
3224          /THE CONTENTS OF THE RST STATUS SHOULD = DRIVE READY (200)
3225

```

```

3226          / (IF UNIT-S WERE SELECTED AT L/S 200 OTHERWISE THE STATUS SHOULD = 0 )
3227
3228 2475 1170          TAD BLANK          / (GOOD) SAVED PRIOR TO ENTRY INTO " XRST "
3229 2476 0367          AND (-4-1)          / MINUS " INIT DONE " BIT
3230 2477 3166          DCA GOOD
3231 2500 1167          TAD EAC
3232 2501 0366          AND (277)          / ACTUAL STATUS MINUS " DELETED DATA "
3233 2502 3170          DCA BLANK
3234 2503 1166          TAD GOOD          /EXPECTED
3235 2504 7041          CIA
3236 2505 1170          TAD BLANK          /ACTUAL (MASSAGED)
3237 2506 7650          SNA CLA
3238 2507 5312          JMP ,+3
3239 2510 1167          TAD EAC          / ACTUAL ACTUAL STATUS
3240 2511 4452          E7005, ERROR          /EXPECTED STATUS NOT = (GOOD)
3241 2512 4502          SCOPE
3242 2513 5641          JMP I XRST
3243
3244          /SUB-TEST SELECTIONS
3245
3246          /
3247          / TEST 0 = " THE TEST "
3248          / TEST 001 = WRITE - READ (PARITY CHECK)
3249          / TEST 010 = WRITE - READ - READ CHECK (PROGRAM VERIFY)
3250          / TEST 011 = READ - READ CHECK (COMPATABILITY)
3251          / TEST 100 = READ (PARITY CHECK)
3252          / TEST 5 = WRITE ONLY
3253          / TEST 110 = EMPTY BUFFER
3254          / TEST 111 =
3255
3256 2514 1365          TEST4, TAD (READ)
3257 2515 7410          SKP
3258 2516 1364          TEST3, TAD (READCOMPARE)
3259 2517 3337          DCA DOB
3260 2520 1375          TAD (NOP)
3261 2521 5331          JMP DCADDA
3262 2522 1364          TEST2, TAD (READCOMPARE)
3263 2523 7410          SKP
3264 2524 1365          TEST1, TAD (READ)
3265 2525 7410          SKP
3266 2526 1375          TEST5, TAD (NOP)
3267 2527 3337          DCA DOB
3268 2530 1363          TAD (WRITE)
3269 2531 3336          DCADDA, DCA DOA
3270
3271 2532 4455          TESTX, GETAPATTERN
3272 2533 4463          INITTRACKS
3273 2534 4460          GETUNIT
3274 2535 4457          TESTXL, GETATRACK
3275 2536 4510          DOA, WRITE
3276 2537 4474          DOB, READ
3277 2540 2147          ISZ TRACKS
3278 2541 5335          JMP TESTXL
3279 2542 4454          DONE
3280 2543 5332          JMP TESTX

```

```

3281      2544 5453      EXIT
3282      /AC SW 7 = 1 TO HALT AT END OF TEST
3283
3284      2545 2545      XLOCKUP, .
3285      2546 4572      LAS
3286      2547 0362      AND (SW7)
3287      2550 7640      SZA CLA
3288      2551 4461      HLT
3289      /AC SW 2 = 1 TO LOCK SCOPE LOOP ON TEST
3290
3291      2552 4572      LAS
3292      2553 0374      AND (SW2)
3293      2554 7640      SZA CLA
3294      2555 5571      JMP I TEST
3295      2556 5745      JMP I XLOCKUP
3296      2562 0020
3297      2563 4510
3298      2564 4475
3299      2565 4474
3300      2566 0277
3301      2567 7773
3302      2570 5000
3303      2571 3000
3304      2572 0012
3305      2573 0377
3306      2574 1000
3307      2575 7000
3308      2576 1165
3309      2577 0016
3310      2600
3311
3312      /
3313      /BYTE SWAP SUBROUTINE
3314      /
3315      /THE CONTENTS OF THE AC AT ENTRY WILL BE SWAPPED
3316      /
3317      XBSW, .
3318      DCA BSWAC
3319      RAR
3320      DCA BSWLINK
3321      TAD (-6)
3322      DCA BSWRAL
3323      TAD BSWAC
3324      CLL
3325      SPA
3326      STL
3327      RAL
3328      ISZ BSWRAL
3329
3330      JMP .-5
3331      DCA BSWAC
3332      TAD BSWLINK
3333      CLL RAL
3334      TAD BSWAC
3335      JMP I XBSW
3336
3337      BSWRAL, -6
3338      BSWAC, 0
    
```

```

3335      2624 0000      BSWLINK, 0
3336      2625 2625      XERROR, .
3337      2626 3167      DCA EAC
3338      2627 1167      TAD EAC
3339      2630 4467      AERROR
3340      2631 2111      ISZ BUSY
3341      2632 1225      TAD XERROR
3342      2633 7001      IAC
3343      2634 3776      DCA EPCSCOPE / ERROR RETURN ADDRESS + 1
3344      2635 7301      CLL CLA IAC
3345      2636 3775      DCA ERRORS
3346      /IF AC SW 4 = 0 THEN PRINT AN ERROR MESSAGE
3347      /
3348      2637 4572      LAS
3349      2640 0374      AND (SW4)
3350      2641 7640      SZA CLA
3351      2642 5315      JMP XNPRINT
3352      /IF THIS IS THE FIRST ERROR FOR THIS PASS THEN PRINT THE HEADER LINE
3353      /
3354      2643 2115      ISZ FIRSTERROR
3355      2644 5247      JMP .+3
3356      2645 4473      PRINT
3357      2646 6545      MXEHEADER
3358      2647 4473      PRINT
3359      2650 6602      MCRLF
3360      /
3361      /
3362      / ERR = ERROR ADDRESS ( E # )
3363      / FAT = FIRST ADDRESS OF TEST
3364      / FAST = FIRST ADDRESS OF SUB-TEST
3365      /
3366      / EAC = CONTENTS OF THE ACCUMULATOR AT THE ERROR
3367      / PASS = PASS IN WHICH ERROR OCCURED
3368      /
3369      2651 7240      STA
3370      2652 1225      TAD XERROR
3371      2653 3160      DCA LSB
3372      2654 4505      TY4OCT
3373      2655 0160      LSB
3374      2656 4504      TAB
3375      2657 0005      5
3376      2660 1171      TAD TEST
3377      2661 3160      DCA LSB
3378      2662 4505      TY4OCT
3379      2663 0160      LSB
3380      2664 4504      TAB
3381      2665 0012      12
3382      2666 4505      TY4OCT
3383      2667 1366      PCSCOPE
3384      2670 4504      TAB
3385      2671 0017      17
3386      2672 4505      TY4OCT
3387      2673 0170      BLANK
3388      2674 4504      TAB
3389      2675 0024      24
    
```

```

3390 2676 4505 TY4OCT
3391 2677 0167 FAC
3392 2700 4504 TAB
3393 2701 0031 31
3394 2702 4505 TY4OCT
3395 2703 0166 GOOD
3396 2704 4504 TAB
3397 2705 0036 36
3398 2706 4434 TY8OCT
3399 2707 0163 PASS
3400 2710 4572 LAS
3401 2711 7700 SMA CLA
3402 2712 5315 JMP ,+3
3403 2713 4504 TAB
3404 2714 0043 43
3405 0207 BELL=207
3406 /
3407 2715 4572 XNOPRINT, LAS
3408 2716 0373 AND (SW11)
3409 2717 7640 SZA CLA
3410 2720 5323 JMP ,+3
3411 2721 4501 SPECIALTYPEIT
3412 2722 0207 BELL
3413 2723 3115 DCA FIRSTERROR
3414 2724 4572 LAS
3415 2725 7700 SMA CLA
3416 2726 4461 HLT
3417 /
3418 2727 3111 DCA BUSY
3419 2730 6001 ION
3420 2731 5625 JMP I XERROR
3421 /
3422 /
3423 /ROUTINE TO DETERMINE WHETHER TO TEST DRIVE ZERO
3424 /OR IF NO DRIVES ARE TO BE TESTED.
3425 /
3426 2732 4772* TSTUNT, JMS VT78CK /GO CHECK IF ON VT78 SYSTEM
3427 2733 4350 JMS CHGMSK /VT78/ GO CHANGE MASKING CONSTANT
3428 2734 1771* TAD UNITS /UNITS SELECTED BY OPERATOR
3429 2735 0370 AND (4000 /ISOLATE DRIVE ZERO.
3430 2736 7640 SZA CLA /IS DRIVE ZERO THERE.
3431 2737 5343 JMP DRVZRO /YES.
3432 2740 1347 TAD MASK1 /DON'T TEST DRIVE ZERO.
3433 2741 3346 DCA CUMP /IT IS NOT ANERROR IF DRIVE
3434 /ZERO IS OPERATIONAL.
3435 2742 5767* JMP NORX01=1
3436 2743 1136 DRVZRO, TAD MASK
3437 2744 3346 DCA CUMP
3438 2745 5766* JMP NORX01=2 /DRIVE ZERO THERE SO INDICATE IT.
3439 /
3440 2746 0000 COMP, 0
3441 2747 7477 MASK1, 7477
3442 /
3443 /ROUTINE TO CHANGE MASKING CONSTANT FOR USE ON VT78
3444 /BITS 0-2 ARE NOT RESIDUAL ON VT78

```

```

3445 /
3446 2750 0000 CHGMSK, 0
3447 2751 1365 TAD (277
3448 2752 3136 DCA MASK
3449 2753 1364 TAD (77
3450 2754 3347 DCA MASK1
3451 2755 5750 JMP I CHGMSK
3452 /
3453 2764 0077
3454 2765 0277
3455 2766 0427
3456 2767 0430
3457 2770 4000
3458 2771 4235
3459 2772 1165
3460 2773 0001
3461 2774 0200
3462 2775 1365
3463 2776 1364
3464 2777 7772
3465 3000 PAGE
3466 3000 4455 /
3467 3001 4463 THETEST, GETAPATTERN
3468 3002 4460 INITTRACKS
3469 GETUNIT
3470 /
3471 /FORCE THE ACTUATOR SEQUENCE = 0 (INCREMENTAL 0=114)
3472 /
3472 3003 1377 XTHEL, TAD (XTHEL)
3473 3004 3776* DCA XGETATRACK
3474 3005 1131 TAD TARGET
3475 3006 3125 DCA START
3476 3007 1150 TAD TTRACKS
3477 3010 1147 TAD TRACKS
3478 3011 7640 SZA CLA
3479 3012 5775* JMP SEQ000
3480 3013 1027 TAD 00
3481 3014 3131 DCA TARGET
3482 3015 5775* JMP SEQ000
3483 3016 4510 XTHEL, WRITE
3484 3017 4475 READCOMPARE
3485 3020 2147 ISZ TRACKS
3486 3021 5203 JMP XTHEL
3487 3022 4454 DONE
3488 3023 5201 JMP THETEST+1
3489 /ACTUATOR MOVEMENT IS THAT SELECTED OF BITS 6,7,8 OF (TESTP)
3490 /
3491 3024 4463 XXTHEL, INITTRACKS
3492 3025 4460 GETUNIT
3493 3026 4457 XYTHEL, GETATRACK
3494 3027 4475 READCOMPARE
3495 3030 2147 ISZ TRACKS
3496 3031 5226 JMP XYTHEL
3497 3032 4454 DONE
3498 3033 5224 JMP XXTHEL

```

```

3499 3034 5453          EXIT
3500                    /GET A PATTERN
3501                    /
3502 3035 3035          XGETAPATTERN, .
3503                    /
3504                    /ONLY GENERATE A PATTERN, HOWEVER, IF THE CONTENTS OF PROGRAM LOCATIONS
3505                    /
3506                    /ERRORS = 0 (MEANS NO ERRORS),
3507                    /
3508                    /AND AC SW2 = 0 (NOT TO LOCK ON TEST OK)
3509                    /
3510                    /AND TEST = T13, T16, T34, T35, T36, T37
3511                    /
3512 3036 4572          LAS
3513 3037 0374          AND (S#2)
3514 3040 1773          TAD ERRORS
3515 3041 7640          SZA CLA
3516 3042 5635          JMP I XGETAPATTERN
3517 3043 1372          TAD (GENTESTS-1)
3518 3044 3012          DCA A12
3519 3045 1412          TAD I A12
3520 3046 7450          SNA
3521 3047 5635          JMP I XGETAPATTERN
3522 3050 1171          TAD TEST
3523 3051 7640          SZA CLA
3524 3052 5245          JMP *-5
3525 3053 1371          TAD (WBUFFER+1)
3526 3054 3010          DCA A10
3527 3055 3155          DCA PATSUMCHECK
3528 3056 7307          CLL CLA IAC RTL
3529 3057 0132          AND TESTP / 8/12 MODE MASK
3530 3060 7640          SZA CLA
3531 3061 1370          TAD (-100) /124 DATA WORDS
3532 3062 1367          TAD (-74)
3533 3063 3011          DCA A11 / 60 DATA WORDS
3534                    /WORD X AND WORD Y CONTAIN THE ADDRESSES OF THE LAST 2 SUMCHECK WORDS
3535                    /
3536 3064 1011          TAD A11
3537 3065 7041          CIA
3538 3066 1366          TAD (WBUFFER+2)
3539 3067 3156          UCA WORDX
3540 3070 7301          CLL CLA IAC
3541 3071 1156          TAD WORDX
3542 3072 3157          UCA WORDY
3543 3073 1365          TAD (7000)
3544 3074 0132          AND TESTP
3545 3075 7106          CLL RTL

3546 3076 7006          RTL
3547 3077 1364          TAD (TAD PATTERNS)
3548 3100 3301          DCA +1
3549 3101 1315          TAD PATTERNS
3550 3102 3303          DCA XPATTERNS
3551 3103 4763          XPATTERNS, JMS RANGEN
3552 3104 3137          DCA XXX
3553                    /DEVELOP A 12 BIT SUMCHECK FOR THE 60/124 DATA WORDS

```

```

3554                    /
3555                    /NOTE:
3556                    /
3557                    /THE SUMCHECK WRITTEN IS THIS DEVELOPED SUMCHECK WITH THE FIRST 2 ID WORDS, AND
3558                    /THE LAST 2 OVERALL SUMCHECK WORDS APPENDED WITHIN SUBROUTINE XGETASECTOR
3559                    /
3560 3105 1155          TAD PATSUMCHECK
3561 3106 1137          TAD XXX
3562 3107 3155          DCA PATSUMCHECK
3563 3110 1137          TAD XXX
3564 3111 3410          DCA I A10
3565 3112 2011          ISZ A11
3566 3113 5303          JMP XPATTERNS
3567 3114 5635          JMP I XGETAPATTERN
3568                    /THE FOLLOWING ARE THE ALLOCATED PATTERN POINTERS
3569                    /
3570 3115 4763          PATTERNS, JMS RANGEN
3571 3116 7240          STA / 1
3572 3117 1325          TAD PAT2
3573 3120 1326          TAD PAT3
3574 3121 1327          TAD PAT4
3575 3122 1330          TAD PAT5
3576 3123 5340          JMP PAT6
3577 3124 7200          CLA
3578                    /
3579                    /
3580                    /
3581 3125 1463          PAT2, 1463 / 63 / 0011 00110011
3582 3126 6314          PAT3, 6314 / 314 / 1100 11001100
3583 3127 5252          PAT4, 5252 / 252 / 1010 10101010
3584 3130 2525          PAT5, 2525 / 125 / 0101 01010101
3585                    /
3586 3131 6322          GENTESTS, -T13
3587 3132 6323          -T16
3588 3133 5500          -T34
3589 3134 5502          -T35
3590 3135 5501          -T36
3591 3136 5503          -T37; 0
3592                    /PROGRAMMING NOTE:
3593                    /
3594                    /PATTERN 6 IS A COUNT PATTERN FROM 0 TO 7777, OR FROM 0 TO 177
3595                    /
3596                    /THIS COUNT PATTERN IS A " PURE " PATTERN
3597                    /
3598                    /A CHECKSUM IS NOT GENERATED FOR THIS PATTERN AS FOR ALL OTHER PATTERNS
3599                    /
3600                    /
3601                    /
3602                    /
3603                    /
3604                    /
3605                    /
3606                    /
3607                    /

```



```

3608 /
3609 3140 7307 PAT6, CLL CLA IAC RTL / 4
3610 3141 0132 AND TESTP
3611 3142 7440 SZA
3612 3143 7120 STL / LINK = 1 FOR 8-BIT MODE (128 BYTES)
3613 3144 5351 JMP .+5
3614 3145 7001 XPAT6, IAC
3615 3146 7420 SNL
3616 3147 1362 TAD (100)
3617 3150 1137 TAD XXX
3618 3151 3137 UCA XXX
3619 3152 1137 TAD XXX
3620 3153 3410 DCA I A10
3621 3154 2011 ISZ A11
3622 3155 5345 JMP XPAT6
3623 3156 5635 JMP I XGETAPATTERN
3624 3162 0100
3625 3163 4712
3626 3164 1315
3627 3165 7000
3628 3166 7116
3629 3167 7704
3630 3170 7700
3631 3171 7115
3632 3172 3130
3633 3173 1365
3634 3174 1000
3635 3175 4306
3636 3176 4261
3637 3177 3016
3200 PAGE
3638 /WRITE ONLY
3639 /
3640 /WRITE ALL SELECTED SECTORS OF THAT TRACK
3641 /
3642 3200 3200 XWRITE,
3643 3201 7330 STL CLA RAR
3644 3202 3120 UCA RUC / THIS IS A WRITE (RDC = 4000)
3645 /
3646 3203 4462 INITSECTORS
3647 3204 4456 WRITE, GETASECTOR
3648 3205 7410 SKP
3649 3206 1377 REWRITE, TAD (JMP WHICHREAD)
3650 3207 3263 UCA JMPWHICHREAD
3651 /THE PROGRAM WILL ISSUE AN INIT FOR ALL SEEK ERRORS
3652 /
3653 /NOTE:
3654 /
3655 /THE FUNCTION OF THE INIT IS TO SEEK TRACK 0/SECTOR 1, AND
3656 /TRANSFER INTO THE SECTOR BUFFER THE CONTENTS OF SECTOR 1, THEREFORE
3657 /TO RECOVER FROM A SEEK ERROR, THE PROGRAM MUST RE-FILL THE SECTOR BUFFER
3658 /THEN RE-SEEK
3659 /
3660 3210 1121 TAD KPRETRY
3661 3211 3145 UCA SRETRY
    
```

```

3662 3212 1121 WRESEEK, TAD KPRETRY
3663 3213 3146 UCA PRETRY /PARITY RETRY COUNTER
3664 /FILL RX01 SECTOR BUFFER
3665 /
3666 3214 1376 REFILL, TAD (WBUFFER-1)
3667 3215 3010 DCA A10
3668 3216 4437 LCDA
3669 3217 5237 JMP FILLOK
3670 3220 5226 JMP FILLERROR
3671 /RETURN TO HERE IS FROM SUBROUTINE XLCDA
3672 /
3673 3221 4443 FILL, STR
3674 3222 5221 JMP FILL
3675 3223 1410 TAD I A10
3676 3224 4442 XDROUT
3677 3225 5221 JMP FILL
3678 /AC SW (4) = 0 ; INHIBIT INTERFACE PARITY RECOVERY
3679 /
3680 3226 2146 FILLERROR, ISZ PRETRY
3681 3227 7410 SKP
3682 3230 5600 JMP I XWRITE
3683 3231 4572 LAS
3684 3232 0375 AND (SW1)
3685 3233 7640 SZA CLA
3686 3234 5214 JMP REFILL
3687 3235 3146 DCA PRETRY
3688 3236 5600 JMP I XWRITE
3689 /THE SECTOR BUFFER HAS SUCCESSFULLY BEEN FILLED AND CONTAINS THE
3690 /PATTERN SELECTED - ALL SOFT PARITY ERRORS HAVE BEEN LOGGED
3691 /
3692 3237 1121 FILLOK, TAD KPRETRY
3693 3240 3146 UCA PRETRY
3694 3241 7305 CLL CLA IAC RAL / 2
3695 3242 0132 AND TESTP /TESTP FOR DELETED DATA SELECTION
3696 3243 7007 IAC RTL / 4 IF WRITE OR 14 IF WRITE DELETED DATA
3697 3244 4440 LCDB
3698 3245 5257 JMP WRITEOK
3699 /RETURN TO HERE IS FROM A PI (IF AN RX01 ERROR FLAG)
3700 /
3701 /AC SW (1) = 0 ; INHIBIT WRITE RETRY
3702 /
3703 3246 2145 WRITERERROR, ISZ SRETRY
3704 3247 7410 SKP
3705 3250 5264 JMP WNOTOK
3706 3251 4572 LAS
3707 3252 0375 AND (SW1)
3708 3253 7640 SZA CLA
3709 3254 5212 JMP WRESEEK
3710 3255 3145 DCA SRETRY
3711 3256 5264 JMP WNOTOK
3712 /NOTE: THE ONLY ERROR EXPECTED HERE EVER IS A "SEEK" ERROR
3713 /
3714 /RETURN TO HERE IS FROM A PI (IF ONLY AN RX01 DONE FLAG)
3715 /
3716 3257 1131 WRITEOK, TAD TARGET
    
```

```

3717 3260 3125          DCA START
3718 3261 1124          TAD STARGET
3719 3262 3123          DCA SSTART
3720 3263 5267          JMPWHICHREAD, JMP WHICHREAD /CONTAINS 0 OR MODIFIED TO JMP WHICHREAD
3721 /
3722 3264 2122          WNOTOK, ISZ SECTORS
3723 3265 5204          JMP WRITEL
3724 3266 5600          JMP I XWRITE
3725 /IF THIS IS A WRITE AFTER READ, "JMP READRETRY", BUT
3726 /
3727 /IF A HARD SEEK ERROR, "JMP NUREAD"
3728 /
3729 3267 1145          WHICHREAD, TAD SRETRY
3730 3270 7650          SMA CLA
3731 3271 5774          JMP NUREAD /HARD SEEK ERROR
3732 3272 5773          JMP READRETRY /SOFT
3733 /
3734 / 0 < (LAST) => (FIRST)
3735 /
3736 3273 1032          TADLAST, TAD LAST
3737 3274 7740          SMA SZA CLA
3738 3275 5300          JMP ,+3
3739 3276 1372          TAD (32)
3740 3277 3032          DCA LAST
3741 3300 1032          TAD LAST
3742 3301 7041          CIA
3743 3302 1031          TAD FIRST
3744 3303 7740          SMA SZA CLA
3745 3304 5276          JMP ,+6
3746 3305 4473          PRINT
3747 3306 6711          MLASI
3748 3307 4505          TY4OCT
3749 3310 0032          LAST
3750 3311 4504          TAB
3751 3312 0005          S /TAB TO MOVE TELEPRINTER HEAD
3752 /*****
3753 /CONSOLE
3754 /*****
3755 /
3756 /
3757 3313 4424          CHECKC8 /IS CONSOLE ACTIVE
3758 3314 4425          XC8ENTR /YES PRINT SWITCH REGISTER QUESTION
3759 /FOR RUNNING SWITCHES
3760 3315 5317          JMP C8RET4
3761 3316 4461          HLT
3762 3317 5771          C8RET4,JMP PRETEST
3763 /ENTRY TO HERE IS FROM RX01 PI ERROR SERVICE ROUTINE

3764 /
3765 /CONTROLLER FAILURES
3766 /
3767 3320 1370          SDNUNEXPECTED, TAD (MSDNUNEXPECTED)
3768 3321 7410          SKP
3769 3322 1367          NUSER, TAD (NUSER)
3770 3323 3334          DCA XMESSAGE
3771 3324 4572          LAS

```

```

3772 3325 0366          AND (SW4)
3773 3326 7640          SZA CLA
3774 3327 5335          JMP QUIET
3775 3330 4576          FORCE
3776 3331 1334          TAD XMESSAGE /POINTS TO ERROR MESSAGE THAT WOULD HAVE BEEN PRINTED,
3777 3332 4467          AERROR /REPORT ERROR TO APT IF REQUIRED.
3778 3333 4473          PRINT
3779 3334 0000          XMESSAGE, 0
3780 3335 4572          QUIET, LAS
3781 3336 7710          SPA CLA
3782 3337 4461          HLT
3783 3340 5765          JMP ERETURN
3784 /
3785 /ENTRY TO HERE FROM SUBROUTINES XSER, OR XSDN
3786 /
3787 3341 2111          HUNGUP, ISZ BUSY
3788 3342 3353          DCA HUNGPC
3789 3343 4576          FORCE
3790 /
3791 /THE LABEL "XHUNG" MUST RESIDE HERE BECAUSE OF REFERENCES MADE WITHIN "FORCE"
3792 /
3793 3344 4473          XHUNG, PRINT
3794 3345 6612          MHUNGPC
3795 3346 4505          TY4OCT
3796 3347 3353          HUNGPC
3797 3350 1364          TAD (-40)
3798 3351 3116          DCA HANGER
3799 3352 5453          NOTEST
3800 3353 0000          HUNGPC, 0
3801 /
3802 3364 7740          /
3803 3365 5520          /
3804 3366 0200          /
3805 3367 7056          /
3806 3370 7043          /
3807 3371 0400          /
3808 3372 0032          /
3809 3373 3425          /
3810 3374 3521          /
3811 3375 2000          /
3812 3376 7113          /
3813 3377 5267          /
3814 /
3815 /ROUTINE IS USED INPLACE OF THE LAS INSTRUCTION WILL READ CORRECT SWITCHES
3816 /
3817 /
3818 /
3819 3400 0000          XCKSWIT, 0
3820 3401 7200          CLA
3821 3402 1022          TAD 22 /CHECK BIT 0 FOR APT
3822 3403 7710          SPA CLA /ON APT?
3823 3404 5600          JMP I XCKSWIT /YES RETURN WITH CLEARED AC
3824 3405 1021          TAD 21 /CHECK STATUS WORD
3825 3406 7710          SPA CLA /0=ACTIVE CONSOLE
3826 3407 7614          TAD 14 /LAS AND SKIP

```

```

3826 3410 1020 TAD 20 /GET PSEUDO SW REG
3827 3411 5600 JMP I XCKSWIT /EXIT XCKSWIT
3828 /READ VERIFY
3829 /
3830 3412 3412 XREADCOMPARE,
3831 3413 1212 TAD XREADCOMPARE
3832 3414 3216 DCA XREAD
3833 3415 5220 JMP XRDC
3834 /READ AND READ AFTER WRITE SUBROUTINE
3835 /
3836 /*****
3837 /
3838 /IF THIS IS A READ AFTER WRITE, THEN 1 PROGRAM LOCATION PRECEEDING
3839 /THE PROGRAM LUCATION CONTAINING THE "READ" WILL CONTAIN "WRITE"
3840 /
3841 /*****
3842 /
3843 /IF THIS IS A READ ONLY, THEN THE CONTENTS OF " RDC " WILL = 1
3844 /
3845 /IF THIS IS A READ VERIFY, THEN THE CONTENTS OF " RDC " WILL = 0
3846 /
3847 /READ ALL SELECTED SECTORS OF THAT TRACK
3848 /
3849 3416 3416 XREAD,
3850 3417 7301 CLL CLA IAC
3851 3420 3120 XRDC, DCA RDC
3852 3421 4462 INITSECTORS
3853 3422 1121 READL, TAD KRETRY
3854 3423 3140 DCA RIRETRY
3855 3424 4456 GETASECTOR
3856 3425 1121 READRETRY, TAD KRETRY
3857 3426 3141 DCA R2RETRY
3858 /REFRESH PROGRAM LOCATION SNDLOG BECAUSE THE CONTENTS MIGHT BE RESIDUAL
3859 /IF A PREVIOUS SND (STATUS NO DATA) ERROR EVER OCCURED
3860 /
3861 3427 3144 DCA SNDLOG
3862 3430 3142 DCA DWSLOG
3863 3431 3143 DCA DNSLOG
3864 /
3865 /
3866 3432 1121 REREAD, TAD KRETRY
3867 3433 3145 DCA SRETRY
3868 3434 1121 RESEEK, TAD KRETRY
3869 3435 3146 DCA PRETRY
3870 3436 7327 CLA STL IAC RTL
3871 3437 4440 LDCR
3872 3440 5331 JMP READOK

3873 /RETURN TO HERE IS FROM A PI (IF AN RX01 ERROR FLAG)
3874 /
3875 /AN ERROR HAS BEEN DETECTED
3876 /
3877 /IF NOT A CRC ERROR THEN ASSUME A SEEK ERROR
3878 /
3879 3441 7301 READERROR, CLL CLA IAC
3880 3442 0126 AND ASTATUS /CRC MASK
    
```

```

3881 3443 7640 SZA CLA
3882 3444 5256 JMP CRCERROR
3883 3445 2145 ISZ SRETRY
3884 3446 7410 SKP
3885 3447 5347 JMP RLOGGED
3886 3450 4572 LAS
3887 3451 0377 AND (SW1)
3888 3452 7640 SZA CLA
3889 3453 5234 JMP RESEEK
3890 3454 3145 DCA SRETRY
3891 3455 5347 JMP RLOGGED
3892 /
3893 /PROGRAMMING NOTE:
3894 /
3895 /ANY RECOVERABLE SEEK ERRORS ARE NOTED AT PROGRAM LOCATION " DNS "
3896 /
3897 /AN ERROR HAS OCCURED
3898 /
3899 3510 SMD=DNS
3900 3510 DWS=SMD
3901 /
3902 /THE SECTOR BUFFER CONTAINS THE DATA READ
3903 /
3904 /ANY PARITY ERRORS WOULD HAVE PREVIOUSLY BEEN DETECTED AND LOGGED
3905 /
3906 /THIS ERROR IS NOT A SEEK ERROR, THEREFORE IT IS ASSUMED TO BE A
3907 /
3908 / CRC ERROR
3909 /
3910 /COMPARE THE DATA WITHIN THE SECTOR BUFFER TO DETECT CRC STATUS
3911 /WITHOUT DATA ERRORS (SND)
3912 /
3913 /IF AT THE END OF THE COMPARE, STATUS NO DATA ERRORS HAVE BEEN DETECTED
3914 /THEN PRINT AN APPROPRIATE MESSAGE
3915 /
3916 3456 1120 CRCERROR, TAD RDC
3917 3457 7640 SZA CLA
3918 3460 5305 JMP XRCERROR
3919 3461 4776 JMS COMPARE
3920 3462 1113 TAD COMPERROR
3921 3463 7450 SWA
3922 3464 1775 TAD INSUMCHECK
3923 3465 7640 SZA CLA
3924 3466 5310 JMP DWS
3925 3467 2144 ISZ SNDLOG
3926 /
3927 /IF AC SW (4) = 1 THEN DO NOT PRINT THE ERROR INFORMATION
3928 /
3929 3470 4572 LAS
3930 3471 0374 AND (SW4)
3931 3472 7640 SZA CLA
3932 3473 5305 JMP XRCERROR
3933 3474 4352 JMS RDDRWR
3934 3475 5301 JMP +4
3935 3476 4473 PRINT
    
```

```

3936 3477 6716 MWRITE
3937 3400 5303 JMP ,+3
3938 3501 4473 PRINT
3939 3502 6722 MREAD
3940 3503 4473 PRINT
3941 3504 6740 MSNDERROR
3942 3405 4572 XRCRCERR, LAS
3943 3506 7700 SMA CLA
3944 3407 4461 HLT7, HLT
3945 /
3946 3510 1121 DNS, TAD RRETRY
3947 3511 3145 DCA RRETRY
3948 3512 2141 ISZ R2RETRY
3949 3513 7410 SKP
3950 3514 5324 JMP UREAD
3951 /AC SW (1) = 0 ; INHIBIT READ RETRY
3952 /
3953 3515 4572 LAS
3954 3516 0377 AND (SW1)
3955 3517 7640 SZA CLA
3956 3520 5232 JMP REREAD
3957 /A HARD FILL BUFFER PARITY ERROP OR A HARD PARITY ERROR ON THE COMMAND/
3958 /SECTOR/TRACK WORDS, OR
3959 /A HARD SEEK ERROR WHICH HAS BEEN LOGGED WITHIN THE WRITE SUBROUTINE, OR
3960 /A HARD CRC ERROP WHICH OCCURED WHILE WITHIN THE READ SUBROUTINE
3961 /
3962 3521 3140 NUREAD, DCA R1RETRY
3963 3522 3141 DCA R2RETRY
3964 3523 5347 JMP RLOGGED
3965 /IF THIS IS A READ AFTER WRITE THEN RE-WRITE THE SECTOR IN ERROR
3966 /
3967 /BECAUSE THIS IS A HARD PROGRAM COMPARE DATA ERROR
3968 /
3969 3524 4352 UREAD, JMS RDORWR
3970 3525 5321 JMP NUREAD
3971 3526 2140 ISZ R1RETRY
3972 3527 5773 JMP RWRITE
3973 3530 5347 JMP RLOGGED
3974 /IF THIS IS A READ COMPARE TEST
3975 /
3976 /THEN " JMS COMPARE ", BUT ...
3977 /
3978 /IF A STATUS ERROR WITH NO DATA ERROR HAS PREVIOUSLY BEEN DETECTED
3979 /
3980 /THEN DO NOT RE- " JMS COMPARE ", AND
3981 /
3982 /DO NOT RESET PROGRAM LOCATION " START " WITH " TARGET "
3983 /
3984 /PROGRAMMING NOTE:
3985 /
3986 /ANY RECOVERABLE SEEK ERRORS WHICH OCCURED PRIOR TO CRC OR DATA ERRORS
3987 /
3988 /ARE LOGGED AT PROGRAM LOCATION " DNS "
3989 /
3990 3531 1144 READOK, TAD SNDLOG
    
```

```

3991 3532 1120 TAD RDC
3992 3533 7640 SZA CLA
3993 3534 5343 JMP ROK
3994 3535 4774 JMS COMPARE
3995 3536 1113 TAD COMPERROR
3996 3537 7450 SNA
3997 3540 1775 TAD INSUMCHECK
3998 3541 7640 SZA CLA
3999 3542 5310 JMP DNS
4000 3543 1131 ROK, TAD TARGET
4001 3544 3125 DCA START
4002 3545 1124 TAD STARGET
4003 3546 3123 DCA SSTART
4004 3547 2122 RLOGGED, ISZ SECTORS
4005 3550 5222 JMP READL
4006 3551 5616 JMP I XREAD
4007 /IF THE PROGRAMMED INSTRUCTION IMMEDIATELY BEFORE " READ "
4008 /IS A " WRITE " , THEN THIS IS A WRITE DATA ERROR, IF NOT,
4009 /THEN THIS IS A READ DATA ERROR
4010 /
4011 /FORM: JMS RDORWR; (READ RETURN); (WRITE RETURN)
4012 /
4013 3552 3552 RDORWR, .
4014 3553 7344 CLL STA RAL
4015 3554 1216 TAD XREAD
4016 3555 3137 DCA XXX
4017 3556 1537 TAD I XXX
4018 3557 1372 TAD (=WRITE)
4019 3560 7650 SNA CLA
4020 3561 2352 ISZ RDORWR
4021 3562 5752 JMP I RDORWR
4022 3572 3270
4023 3573 3206
4024 3574 0200
4025 3575 3762
4026 3576 3601
4027 3577 2000
4028 3600 5601 PAGE
JMPICOMPARE, JMP I COMPARE
4029 /
4030 /THE FOLLOWING INFORMATION IS ALWAYS PRINTED IF A PROGRAM COMPARE DATA ERROR
4031 /
4032 / DATA ERROR
4033 / WORD GOOD BAD
4034 /
4035 /WHERE " WORD " IS THE WORD NUMBER (0-127),
4036 /AND " GOOD " IS THE DATA WORD WRITTEN,
4037 /AND " BAD " IS THE DATA WORD READ (IN ERROR)
4038 /
4039 /WORDS 0 AND 1 ARE HEADER WORDS (8 BIT BYTES)
4040 /
4041 / WORD 0 = TRACK # (BITS 5-11)
4042 / WORD 1 = SECTOR (BITS 7-11)
4043 /
4044 /
    
```

```

4045          /AC SW (4) = 1 ; INHIBIT READ DATA ERROR TYPEOUT
4046          /
4047          /
4048 3601 3601 COMPARE, .
4049          /
4050          /EMPTY BUFFER
4051          /
4052          /SAVE THE PREVIOUS READ'S A-STATUS REGISTER
4053          /
4054 3602 7301          CLL CLA IAC
4055 3603 0126          AND ASTATUS
4056 3604 3361          DCA XASTATUS
4057 3605 1121          TAD KRETRY
4058 3606 3146          DCA PRETRY
4059 3607 3362          DCA INSUMCHECK
4060 3610 4777          JMS EMPTYL
4061 3611 1132          TAD TESTP
4062 3612 0376          AND (4
4063 3613 7640          SZA CLA          /IS IT 12 BIT MODE?
4064 3614 1375          TAD (7400          /NO (=377)
4065 3615 7040          CMA          /YES
4066 3616 3363          DCA BITMODE          /SAVE MASK
4067          /
4068          /AC SW (4) = 1 ; INHIBIT FURTHER READ DATA ERROR TYPEOUTS
4069          /
4070          /...BUT STAY IN THIS LOOP UNTIL THE RX01 SECTOR BUFFER IS EMPTIED
4071          /
4072 3617 4443 EMPTY, STP
4073 3620 5217 JMP EMPTY
4074          /
4075          /DISABLE THE RX01 INIERRUPT
4076          /
4077 3621 7300          CLA CLL
4078 3622 6756 K67X6B, 6756
4079          /
4080          /TRANSFER DATA FROM THE SECTOR BUFFER INTO THE ACCUMULATOR
4081          /
4082 3623 6752 K67X2C, 6752          / "ACTUAL" (BAD) DATA
4083 3624 3133 DCA XA10
4084          /
4085          /ASSUME A 12 BIT SUMCHECK EVEN IF 8 BIT MODE FOR NOW
4086          /
4087          /NOTE: THE CONTENTS OF PROGRAM LOCATION " INSUMCHECK " SHOULD = 0
4088          / (AFTER ALL WORDS HAVE COME IN)
4089          /
4090 3625 1133 TAD XA10
4091 3626 1362 TAD INSUMCHECK
4092 3627 3362 DCA INSUMCHECK
4093 3630 1534 TAD I XA11
4094 3631 0363 AND BITMODE          /377 FOR 8 BIT 7777 FOR 12 BIT
4095 3632 7041 CIA
4096 3633 1133 TAD XA10
4097 3634 7640 SZA CLA
4098 3635 5243 JMP DATAERR
4099 3636 7301 ENDCOMPARE, CLA CLL IAC

```

```

4100 3637 6756 K67X6C, 6756
4101 3640 2134 ISZ XA11
4102 3641 6001 ION
4103 3642 5217 JMP EMPTY
4104          /
4105          /A PROGRAM COMPARE DATA ERROR HAS BEEN DETECTED
4106          /
4107          /NOTATION:
4108          /
4109          /THE CONTENTS OF PROGRAM LOCATION XASTATUS REFLECTS THE STATUS OF THE
4110          /RX01 AT THE COMPLETION OF THE PREVIOUS " READ SECTOR "
4111          /
4112          /IF A CRC ERROR EXISTS THEN THIS IS A DATA ERROR WITH STATUS ERROR
4113          /
4114          /IF A CRC ERROR DOESN'T EXIST THEN THIS IS A DATA ERROR WITH NO ERROR STATUS
4115          /
4116 3643 1361 DATAERR, TAD XASTATUS          / A-STATUS OF PREVIOUS READ
4117 3644 7450 SNA
4118 3645 2143 ISZ DNSLOG          /NO CRC STATUS
4119 3646 7640 SZA CLA
4120 3647 2142 ISZ DNSLOG          /CRC STATUS
4121          /
4122          /AC SW (4) TO INHIBIT ERROR PRINTOUT
4123          /
4124          /IF THIS IS A READ ONLY TEST (NOT READ COMPARE) OR IF AC SW 4 = 1
4125          /THEN DO NOT PRINT ERROR INFORMATION
4126          /
4127          /BUT,
4128          /
4129          /IF THIS IS A READ COMPARE TEST AND AC SW 4 = 0
4130          /THEN PRINT ERROR INFORMATION
4131          /
4132 3650 4572 LAS
4133 3651 0374 AND (SW4)
4134 3652 1120 TAD RDC
4135 3653 7640 SZA CLA
4136 3654 5357 JMP ISZCOMPERROR
4137          /IF THIS IS THE FIRST COMPARE ERROR THEN PRINT HEADER INFORMATION
4138          /
4139 3655 1113 TAD COMPERROR
4140 3656 7640 SZA CLA
4141 3657 5320 JMP CNOTFIRST
4142          /
4143          /IF A DATA NO CRC STATUS ERROR (DNS) THEN PRINT ENTIRE ERROR INFORMATION
4144          /
4145 3660 1143 TAD DNSLOG
4146 3661 7640 SZA CLA
4147 3662 4576 FORCE
4148          /IF THE PROGRAMMED INSTRUCTION IMMEDIATELY BEFORE " READ "
4149          /IS A " WRITE " , THEN THIS IS A WRITE DATA ERROR, IF NOT,
4150          /THEN THIS IS A READ DATA ERROR
4151          /
4152 3663 4773 XCOMPARE, JMS RDORWR
4153 3664 5270 JMP ,+4
4154 3665 4473 PRINT

```

```

4155 3666 6716 MWRITE
4156 3667 5272 JMP ,+3
4157 3670 4473 PRINT
4158 3671 6722 MREAD
4159 /NOTATION:
4160 /
4161 /THE CONTENTS OF PROGRAM LOCATION XASTATUS REFLECTS THE STATUS OF THE
4162 /RX01 AT THE COMPLETION OF THE PREVIOUS " READ SECTOR "
4163 /
4164 /IF A CRC ERROR EXISTS THEN THIS IS A DATA ERROR WITH STATUS ERROR
4165 /
4166 /IF A CRC ERROR DOESN'T EXIST THEN THIS IS A DATA ERROR WITH NO ERROR STATUS
4167 /
4168 3672 1361 TAD XASTATUS
4169 3673 7640 SZA CLA
4170 3674 5300 JMP DWESERROR
4171 3675 4473 PRINT
4172 3676 6726 MDNSERROR
4173 3677 5302 JMP ,+3
4174 3700 4473 DWESERROR, PRINT
4175 3701 6734 MDWESERROR
4176 3702 4473 PRINT
4177 3703 6753 MDATAERROR
4178 /PRINT " WORD " IF 12-BIT MODE, OR PRINT " BYTE " IF 8-BIT MODE
4179 /
4180 3704 1372 TAD (MWORD)
4181 3705 3315 DCA XLENGTH
4182 3706 7307 CLL CLA IAC RTL
4183 3707 0132 AND TESTP
4184 3710 7650 SNA CLA
4185 3711 5314 JMP ,+3
4186 3712 1371 TAD (MBYTE)
4187 3713 3315 DCA XLENGTH
4188 3714 4473 PRINT
4189 3715 6762 XLENGTH, MWORD
4190 3716 4473 PRINT
4191 3717 6770 MGB
4192 / AC 5 = 0 - PRINT ONLY FIRST 3 COMPARE ERRORS
4193 / AC 5 = 1 - PRINT ALL COMPARE ERRORS
4194 /
4195 3720 4572 CNOTFIRST, LAS
4196 3721 0370 AND (SWS)
4197 3722 7640 SZA CLA
4198 3723 5330 JMP ,+5
4199 3724 7346 CLL STA RTL
4200 3725 1113 TAD COMPREPROR
4201 3726 7700 SMA CLA
4202 3727 5357 JMP ISZCOMPREPROR
4203 3730 4473 PRINT
4204 3731 6602 MCRLF
4205 3732 1367 TAD (-WBUFFER)
4206 3733 1134 TAD XA11
4207 3734 3160 DCA LSB
4208 3735 4505 TY4OCT
4209 3736 0160 LSB

```

```

4210 3737 4504 TAB
4211 3740 0005 S
4212 /IF 8-BIT MODE THEN MASK THE 12-BIT "GOOD" WORDS (377)
4213 /
4214 3741 7307 CLL CLA IAC RTL
4215 3742 0132 AND TESTP
4216 3743 7112 CLL RIR
4217 3744 7010 RAR
4218 3745 1534 TAD I XA11
4219 3746 7430 SZL
4220 3747 0366 AND (377)
4221 3750 3160 DCA LSB
4222 3751 4505 TY4OCT
4223 3752 0160 LSB
4224 3753 4504 TAB
4225 3754 0012 12
4226 3755 4505 TY4OCT
4227 3756 0133 XA10
4228 3757 2113 ISZCOMPREPROR, ISZ COMPREPROR
4229 3760 5236 JMP ENDCOMPARE
4230 /THE CONTENTS OF THE XASTATUS PROGRAM LOCATION REFLECTS THE STATUS AT THE
4231 /COMPLETION OF THE PREVIOUS READ FUNCTION
4232 /
4233 /THE PREVIOUS STATUS IS SAVED BECAUSE A SUCCEEDING ERROR
4234 / (PARITY ERROR ON THE COMMAND WORD) MAY OCCUR
4235 /
4236 3761 0000 XASTATUS, 0
4237 /
4238 /THE CONTENTS OF " INSUMCHECK " IS THE SUMCHECK DERIVED FROM THE SUM OF ALL
4239 /DATA WORDS COMING IN (THE SUM OF ALL " BAD " ), AND
4240 /SHOULD BE EQUIVALENT TO 0 AT THE END OF THE EMPTY BUFFER
4241 /
4242 3762 0000 INSUMCHECK, 0
4243 3763 0000 BITMODE, 0
4244 3766 0377
4245 3767 0664
4246 3770 0100
4247 3771 6765
4248 3772 6762
4249 3773 3552
4250 3774 0200
4251 3775 7400
4252 3776 0004
4253 3777 4000
4254 / PAGE
4255 4000 0000 EMPTYL, 0
4256 4001 1377 TAD (WBUFFER)
4257 4002 3134 DCA XA11
4258 4003 7305 CLL CLA IAC RAL
4259 4004 4437 LCDA
4260 4005 5222 JMP EMPIYOK
4261 4006 5211 JMP EMPIYERROR
4262 /RETURN TO HERE FROM SUBROUTINE XLCDA
4263 /

```

```

4264 407 3113          DCA COMPERROR
4265 4010 5600         JMP I EMPTYL          /RETURN
4266
4267 //THIS INTERFACE PARITY ERROR MUST BE ON THE COMMAND WORD TO "EMPTY BUFFER"
4268 //
4269 //NOTE: IT CAN'T BE ANY OTHER ERROR
4270 //
4271 4011 2146         EMPTYERROR, ISZ PRETRY
4272 4012 7410         SKP
4273 4013 5272         JMP XEMPTYOKNOTOK
4274 4014 4572         LAS
4275 4015 0376         AND (SW1)
4276 4016 7640         SZA CLA
4277 4017 5200         JMP EMPTYL
4278 4020 3146         DCA PRETRY
4279 4021 5272         JMP XEMPTYOKNOTOK
4280 //IF AC SW (4) = 0
4281 //
4282 //PRINT A VALUE SYMBOLIC OF THE TOTAL # OF COMPARE ERRORS DETECTED
4283 //
4284 //IF 8-BIT MODE THEN THE "SUMCHECK" WILL OVERFLOW INTO BITS 0 TO 3
4285 //
4286 //THEREFORE MASK THE CONTENTS OF " INSUMCHECK "
4287 //
4288 //THE RESULT OF THE SUBTRACTION SHOULD = 0
4289 //
4290 //NO MASK IS NEEDED FOR 12-BIT MODE
4291 //
4292 4022 7307         EMPTYOK, CLL CLA IAC RIL
4293 4023 0132         AND TESTP
4294 4024 7112         CLL RIR
4295 4025 7010         RAR
4296 4026 1775*      TAD INSUMCHECK
4297 4027 7430         SZL
4298 4030 0374         AND (377)
4299 4031 3775*      DCA INSUMCHECK
4300 4032 4572         LAS
4301 4033 0373         AND (SW4)
4302 4034 7640         SZA CLA
4303 4035 5272         JMP XEMPTYOK
4304 //IF A SUMCHECK ERROR EXISTS WITHOUT A "COMPERROR" THEN "FORCE" A TYPEOUT
4305 //
4306 4036 1113         TAD COMPERROR
4307 4037 7640         SZA CLA
4308 4040 5245         JMP ,+5
4309 4041 1775*      TAD INSUMCHECK
4310 4042 7650         SNA CLA
4311
4312 4043 5272         JMP XEMPTYOK
4313 4044 4576         FORCE
4314 4045 4473         PRINT
4315 4046 6775         MSUMCHECK
4316 4047 1775*      TAD INSUMCHECK
4317 4050 7640         SZA CLA
4318 4051 5255         JMP ,+4
4319 4052 4473         PRINT

```

```

4319 4053 7005         MOK
4320 4054 5260         JMP ,+4
4321 4055 4505         TI40CT
4322 4056 3762         INSUMCHECK
4323 4057 5263         JMP ,+4
4324 4060 1113         TAD COMPERROR
4325 4061 7650         SNA CLA
4326 4062 5272         JMP XEMPTYOK
4327 4063 4473         PRINT
4328 4064 7007         MDESUMMARY
4329 4065 4505         TI40CT
4330 4066 0113         COMPERROR
4331 4067 4504         TAB
4332 4070 0005         5
4333 4071 5272         JMP XEMPTYOK
4334
4335 //AC SW 0 = 1 ; HALT ON ERROR
4336 //
4337 4072 1113         XEMPTYOK, TAD COMPERROR
4338 4073 7450         SNA
4339 4074 1775*      TAD INSUMCHECK
4340 4075 7650         SNA CLA
4341 4076 5772*      JMP JMPICOMPARE
4342 4077 4572         LAS
4343 4100 7710         SPA CLA
4344 4101 4461         HLT6, HLT
4345 4102 5772*      JMP JMPICOMPARE
4346 4103 4103         XSETUP,
4347 4104 4507         WAITTY
4348 4105 1371         TAD (ANDRETURN)
4349 4106 3770*      DCA XPRINT
4350 4107 1703         TAD I XSETUP
4351 4110 3320         DCA XANDRETURN
4352 4111 2303         ISZ XSETUP
4353 4112 5703         JMP I XSETUP
4354
4355 //AND RETURN TO HERE WAITING FOR ENTIRE NUMERICAL OUTPUT
4356 //
4357 4113 4507         ANDRETURN, WAITTY
4358 //
4359 //THEN EXIT FROM HERE FOR MAIN LINE CODE
4360 //
4361 4114 1720         THENEXIT, TAD I XANDRETURN
4362 4115 3320         DCA XANDRETURN
4363 4116 3162         DCA ITYBUSY
4364 4117 5720         JMP I XANDRETURN
4365 4120 4120         XANDRETURN,
4366 //
4367 4121 4121         XHALT,
4368 4122 4507         WAITTY
4369 //*****
4370 //CONSOLE
4371 //*****
4372 4123 4424         CHECKCB
4373 4124 4425         XCENTR

```

```

4374 4125 5331      JMP C8RET3      /
4375                /*****
4376 4126 7240      STA
4377 4127 1321      TAD XHALT
4378 4130 7402      7402
4379 4131 7200      C8RET3,CLA
4380 4132 5721      JMP I XHALT
4381                /
4382                /ROUTINE TO NOTIFY OF OF RUNNING IF NEED BE DONE
4383                /
4384 4133 0000      XTICK, 0
4385 4134 4472      CHEK22
4386 4135 7410      SKP                      /ON APT.
4387 4136 5733      JMP I XTICK
4388 4137 2351      LSZ CLKCNT
4389 4140 5733      JMP I XTICK
4390 4141 1352      TAD COUNT
4391 4142 3351      DCA CLKCNT                      /INIT CLOCK COUNTER
4392 4143 6002      IOF
4393 4144 6201      CDF 00
4394 4145 6272      CIF 70
4395 4146 4750      JMS I K6500                      /NOTIFY API-8
4396 4147 5733      JMP I XTICK                      /EXIT.
4397                /
4398 4150 6500      K6500, 6500
4399 4151 7777      CLKCNT, -1
4400 4152 7763      COUNT, -15
4401 4170 5244
4402 4171 4113
4403 4172 3600
4404 4173 0200
4405 4174 0377
4406 4175 3762
4407 4176 2000
4408 4177 7114
4409                PAGE
4410                /SEQUENCE TO THE NEXT AVAILABLE DISKETTE
4411                /
4411 4200 4200      XGETUNIT,
4412 4201 3240      DCA UNITZ                      /CLEAR FOR A NEW DISKETTE
4413 4202 1236      TAD WUNITS                      /WORKING UNIT COUNTER
4414 4203 7450      SNA
4415 4204 1235      TAD UNITS                      /EXHAUSTED ALL DISKETTES ; RESET
4416 4205 3236      DCA WUNITS
4417 4206 3237      DCA POLL                      /CLEAR POLLER
4418 4207 7120      STL                          /START
4419 4210 1237      NEXT, TAD POLL
4420 4211 7010      RAR                      /NEXT
4421 4212 3237      DCA POLL
4422 4213 1237      TAD POLL
4423 4214 7430      SZL                      /DISKETTE
4424 4215 4461      HLT                      /NO UNITS AVAILABLE ; CATASTROPHIC
4425 4216 0236      AND WUNITS
4426 4217 7450      SNA
4427 4220 2240      ISZ UNITZ                      /ACTIVE DISKETTE

```

```

4428 4221 7450      SNA
4429 4222 5210      JMP NEXT                      /TRY AGAIN
4430                /A DISKETTE IS AVAILABLE AND SELECTED FOR OPERATIONS
4431 4223 3241      DCA UNITX                      /A CODED VERSION OF UNIT
4432 4224 1241      TAD UNITX
4433 4225 7040      CMA                      /...DELETE FROM
4434 4226 0236      AND WUNITS                      /...AVAILABLE UNIT LIST (WUNITS)
4435 4227 3236      DCA WUNITS                      /...AND CREATE NEW LIST
4436 4230 1240      TAD UNITZ
4437 4231 7106      CLL RTL
4438 4232 7006      RTL                      /BIT 7 OF COMMAND REGISTER
4439 4233 3242      DCA UNIT                      /FOR COMMAND REGISTER LOAD LATER
4440 4234 5600      JMP I XGETUNIT
4441 4235 6000      UNITS, 6000                      /AVAILABLE UNIT LIST (MAX SYS CONFIGURATION)
4442 4236 0000      WUNITS, 0                      /CODED WORKING UNIT LIST (UNITS YET TO BE EXERCISED)
4443 4237 0000      POLL, 0                      /DISKETTE POLL
4444                /
4445                /UNITZ ; UNIT 1 LOOKS LIKE 0001
4446                /UNITX ; UNIT 1 LOOKS LIKE 2000
4447                /UNIT ; UNIT 1 LOOKS LIKE 0020 (RX01 COMMAND WORD BIT 7)
4448                /
4449 4240 0000      UNITZ, 0                      /ACTIVE DISKETTE
4450 4241 0000      UNITX, 0                      /ACTIVE DISKETTE IN CODED FORM
4451 4242 0000      UNIT, 0                      /A CODED VERSION OF UNIT Z (FOR COMMAND REGISTER LOAD)
4452                /
4453                /IF (WUNITS) = 0 ; ALL SELECTED DISKETTE DRIVES HAVE SEQUENCED
4454                /
4455 4243 4243      XDONE,
4456 4244 1236      TAD WUNITS
4457 4245 7650      SNA CLA
4458 4246 2243      ISZ XDONE                      / DONE ; (WUNITS) = 0
4459 4247 5643      JMP I XDONE
4460                /INITIALIZE THE NUMBER OF TRACKS ACCESSED VIA THE DIFFERENCE BETWEEN
4461                /THE CONTENTS OF PROGRAM LOCATIONS " OD " AND " ID ".
4462                /
4463 4250 4250      XINITTRACKS,
4464 4251 1030      TAD ID
4465 4252 7040      CMA
4466 4253 1027      TAD OD
4467 4254 3147      DCA TRACKS
4468 4255 1147      TAD TRACKS
4469 4256 7041      CIA
4470 4257 3150      DCA ITRACKS
4471 4260 5650      JMP I XINITTRACKS
4472                / OD (OUTSIDE DIAMETER) ACTUATOR POSITION INITIALIZED TO (0)
4473                /
4474                / ID (INSIDE DIAMETER) ACTUATOR POSITION INITIALIZED TO 114 (76 DECIMAL)
4475                /
4476                / " XGETATRACK " WILL GET A TRACK VALUE BETWEEN THE LIMITS OF THE CONTENTS OF
4477                /PROGRAM LOCATION OD (MIN 0), AND THE CONTENTS OF ID (MAX 114).
4478                /
4479                /GET A DISKETTE TRACK TO BE (AC) WITHIN IOT LCD-B
4480                /
4481                / IF THIS IS THE " FIRSTMOVE ", (START) IS NOT APPLICABLE HERE, BUT
4482                / WILL BECOME APPLICABLE WITHIN SUBROUTINE " XGETASECTOR ".

```



```

4483 /
4484 4761 4261 XGETATRACK,
4485 4762 1131 TAD TARGET
4486 4763 3125 DCA START /PRESENT ACTUATOR POSITION (FROM PREVIOUS LCD-B)
4487 4264 1377 TAD (70) /BECOMES STARTING ACTUATOR POSITION
4488 4765 0132 AND TESTP
4489 4266 7110 CLL KAR
4490 4767 7012 RTR
4491 4270 1376 TAD (TAD SEQ)
4492 4271 3272 DCA ,+1
4493 4272 1276 TAD SEQ
4494 4273 3275 DCA ,+2
4495 4274 5675 RESEQUENCE, JMP I ,+1
4496 4275 4306 SEQ000
4497 /TRACK ACCESS SEQUENCE IS SELECTED VIA AC SWITCHES 6,7,8 AT L/S 200
4498 /
4499 / 0 = INCREMENTAL (1-114=0)
4500 / 1 = DECREMENTAL (114=0)
4501 / 2 = 1-114, 113=0
4502 / 3 = BOUNCE ID TO OD ONLY
4503 / 4 = BOUNCE: (114, 0 ; 113, 1 ; 112, 2 ; ...ETC TO 47, 45)
4504 / 5 =
4505 / 6 = STROBE: (77, 0 ; 76, 0 ; 75, 0 ; ...ETC TO 1, 0)
4506 / 7 = RANDOM
4507 /
4508 4276 4306 SEQ, SEQ000
4509 4277 4317 SEQ001
4510 4300 4400 SEQ010
4511 4301 4422 SEQ3
4512 4302 4444 SEQ100
4513 4303 5453 NOTEST
4514 4304 4465 SEQ6
4515 4305 4330 SEQ111
4516 /
4517 /INCREMENTAL ACTUATOR ACCESS (OD INCREMENTALLY TO ID)
4518 /
4519 4306 1150 SEQ000, TAD TTRACKS
4520 4307 1147 TAD TTRACKS
4521 4310 7640 SZA CLA
4522 4311 5314 JMP ,+3
4523 4312 1027 TAD OD
4524 4313 5332 JMP DCATARGET
4525 4314 1131 TAD TARGET
4526 4315 7001 IAC
4527 4316 5332 JMP DCATARGET
4528 /DECREMENTAL ACTUATOR ACCESS (ID TO OD)
4529 /
4530 4317 1150 SEQ001, TAD TTRACKS
4531 4320 1147 TAD TTRACKS
4532 4321 7640 SZA CLA
4533 4322 5325 JMP ,+3
4534 4323 1030 TAD ID
4535 4324 5332 JMP DCATARGET
4536 4325 1131 TAD TARGET
4537 4326 1117 TAD K7777
    
```

```

4538 4327 5332 JMP DCATARGET
4539 /RANDOM ACTUATOR ACCESS
4540 /
4541 4330 4775 SEQ111, JMS RANGEN
4542 4331 0374 AND (177)
4543 4332 3131 DCATARGET, DCA TARGET
4544 /PREVIOUSLY AT L/S THE FOLLOWING CONDITION WAS TRUE
4545 /
4546 / 0 <= (OD) <= (ID) <= 32
4547 /
4548 /THEREFORE TEST FOR THE CONDITION
4549 /
4550 / (TARGET) <= (ID)
4551 /
4552 4333 1030 XSEQ, TAD ID
4553 4334 7041 CIA
4554 4335 1131 TAD TARGET
4555 4336 7740 SMA SZA CLA
4556 4337 5274 JMP RESEQUENCE
4557 /AND TEST FOR THE CONDITION
4558 /
4559 / (OD) <= (TARGET)
4560 /
4561 4340 1131 TAD TARGET
4562 4341 7041 CIA
4563 4342 1027 TAD OD
4564 4343 7740 SMA SZA CLA
4565 4344 5274 JMP RESEQUENCE
4566 4345 5661 JMP I XGETATRACK
4567 /
4568 /ERROR REPORTER FOR APT, INDICATES TEST PC IN ERROR, THE ONLY EXCEPTIONS
4569 /ARE FOR AN UNEXPECTED RX01 INTERRUPT,
4570 /AND A MISSING ERROR FLAG,
4571 /
4572 4346 0000 XAERRO, 0
4573 4347 3137 DCA XXX /STORE ERROR PC.
4574 4350 4472 CHEK22
4575 4351 7410 SKP
4576 4352 5746 JMP I XAERRO
4577 4353 6002 IOF
4578 4354 1137 TAD XXX /GET BACK ERROR PC.
4579 4355 6201 CDF 00
4580 4356 6272 CIF 70
4581 4357 5761 JMP I K6520
4582 4360 4461 HLT /SOMETHING WENT WRONG ON
4583 /ON REPORTING ERROR TO APT
4584 4361 6520 K6520, 6520
4585 4374 0177
4586 4375 4712
4587 4376 1276
4588 4377 0070
4589 / PAGE
4590 / OD+1 INCREMENTALLY TO ID; ID-1 DECREMENTALLY TO OD
4591 4400 1150 SEQ010, TAD TTRACKS
    
```

```

4592 4401 1147 TAD TRACKS
4593 4402 7640 SZA CLA
4594 4403 5213 JMP XSEQ2
4595 4404 1147 TAD TRACKS
4596 4405 7104 CLL RAL
4597 4406 1377 TAD (2)
4598 4407 3147 DCA TRACKS / (TRACKS X 2)-2 = 152(MAX VAL)
4599 4410 1147 TAD TRACKS
4600 4411 7041 CIA
4601 4412 3150 DCA TTRACKS
4602 /INCREMENT IF (TRACKS) < [ (ID)-(OD) ]
4603 /DECREMENT IF (TRACKS) => [ (ID)-(OD) ]
4604 4413 1027 XSEQ2, TAD OD
4605 4414 7041 CIA
4606 4415 1030 TAD ID
4607 4416 1147 TAD TRACKS
4608 4417 7700 SMA CLA
4609 4420 5776* JMP SEQ001 / ID-1 TO OD
4610 4421 5775* JMP SEQ000 / OD+1 TO ID
4611 /
4612 /BOUNCE; ID,OD ONLY
4613 /
4614 4422 1150 SEQ3, TAD TTRACKS
4615 4423 1147 TAD TRACKS
4616 4424 7640 SZA CLA
4617 4425 5233 JMP XSEQ3
4618 4426 7344 CLL STA RAL
4619 4427 3147 DCA TRACKS
4620 4430 1147 TAD TRACKS
4621 4431 7041 CIA
4622 4432 3150 DCA TTRACKS
4623 4433 1131 XSEQ3, TAD TARGET
4624 4434 7041 CIA
4625 4435 1027 TAD OD
4626 4436 7650 SNA CLA
4627 4437 1030 TAD ID
4628 4440 7450 SNA
4629 4441 1027 TAD OD
4630 4442 3131 DCA TARGET
4631 4443 5774* JMP XSEQ
4632 /BOUNCE ; ID TO OD
4633 /
4634 /THEREFORE 76 (NOT 77) TRACKS ARE TESTED
4635 /
4636 4444 1150 SEQ100, TAD TTRACKS
4637 4445 1147 TAD TRACKS
4638 4446 7640 SZA CLA
4639 4447 5254 JMP Q4UD
4640 /FIRST ENTRY INTO SEQUENCE # 4
4641 /
4642 /IF (TRACKS) = [ (ID)-(OD) ] IS A NEGATIVE ODD THEN INCREMENT (TRACKS)
4643 /
4644 4450 7201 CLA IAC
4645 4451 0147 AND TRACKS
4646 4452 7640 SZA CLA
    
```

```

4647 4453 2147 ISZ TRACKS
4648 /IF (TRACKS) = -ODD ; BOUNCE TO OUTSIDE DIAMETER, BUT
4649 /
4650 /IF (TRACKS) = -EVEN ; BOUNCE TO INSIDE DIAMETER
4651 /
4652 4454 7201 Q4UD, CLA IAC
4653 4455 0147 AND TRACKS
4654 4456 7650 SNA CLA
4655 4457 5305 JMP Q46ID
4656 4460 7240 STA
4657 4461 1150 TAD TTRACKS
4658 4462 1147 TAD TRACKS
4659 4463 7110 CLL RAR
4660 4464 5313 JMP Q46OD
4661 // (TRACKS) = -EVEN ; THEREFORE THE BOUNCE IS TO THE INSIDE DIAMETER
4662 //
4663 /Q4ID, TAD TTRACKS; TAD TRACKS; CLL RAR; CIA; TAD ID; DCA TARGET; JMP XSEQ
4664 /STROBE ID, OD ; ID-1, OD ; ID=2, OD ; ...ETC...
4665 /
4666 4465 1150 SEQ6, TAD TTRACKS
4667 4466 1147 TAD TRACKS
4668 4467 7640 SZA CLA
4669 4470 5300 JMP Q6OD
4670 /FIRST ENTRY INTO SEQUENCE # 6
4671 /
4672 4471 1147 TAD TRACKS
4673 4472 7105 CLL IAC RAL
4674 4473 3147 DCA TRACKS
4675 4474 1147 TAD TRACKS
4676 4475 7041 CIA
4677 4476 3150 DCA TTRACKS
4678 4477 5305 JMP Q46ID
4679 /
4680 / IF (TARGET) = (OD), THEN STROBE TO INSIDE DIAMETER
4681 /
4682 4500 1131 Q6OD, TAD TARGET
4683 4501 7041 CIA
4684 4502 1027 TAD OD
4685 4503 7640 SZA CLA
4686 4504 5313 JMP Q46OD
4687 /
4688 /
4689 /
4690 4505 1150 Q46ID, TAD TTRACKS
4691 4506 1147 TAD TRACKS
4692 4507 7110 CLL RAR
4693 4510 7041 CIA
4694 4511 1030 TAD ID
4695 4512 7410 SKP
4696 4513 1027 Q46OD, TAD OD
4697 4514 3131 DCA TARGET
4698 4515 5774* JMP XSEQ
4699 /*****
4700 /CONSOLE
4701 /*****
    
```

```

4702
4703
4704 4516 0000 XCHECKC8, 0
4705 4517 7200 CLA /CHECK IF CONSOL IS ACTIVE
4706 4520 1022 TAD 22
4707 4521 0373 AND (400
4708 4522 7650 SNA CLA
4709 4523 5337 JMP NOTCL8 /NOT ON ACTIVE CONSOLE
4710 4524 1716 TAD I XCHECKC8 /GET CONSOLE CALL
4711 4525 3331 DCA PACKDD /STORE IT IN THIS LOC
4712 4526 4507 WAITTY /WAIT FOR PRINTING TO FONISH
4713 4527 6002 IOF /KILL INTERRUPT
4714 4530 6212 CIF 10 /CHANGE INST FIELD TO FIELD ONE
4715 4531 0000 PACKDD, 0000 /DO CONSOLE CALL
4716 4532 7000 NOP
4717 4533 6001 ION /RETURN FROM CALL
4718 4534 2316 EXITCK, ISZ XCHECKC8 /BUMP RETURN
4719 4535 7200 CLA
4720 4536 5716 JMP I XCHECKC8 /EXIT XCHECK
4721 4537 2316 NOTCL8, ISZ XCHECKC8 /BUMP FOR NOT CONSOL CALL +2
4722 4540 5334 JMP EXITCK /LEAVE BY THIS MEANS
4723
4724
4725 /*****
4726 /CONSOLE
4727 /*****
4728 /THIS SUBROUTINE IS ENTERED FROM TESTS: T21, T22, T32, T33
4729 /
4730 /THE CONTENTS OF THE AC AT ENTRY IS THE PATTERN TO FILL THE BUFFER WITH
4731 /
4732 4541 4541 FB128BYTES, .
4733 4542 3166 DCA GOOD /THE CONTENTS OF GOOD IS THE PURE PATTERN
4734 4543 1372 TAD (100)
4735 4544 4436 LCD /FILL BUFFER (8-BIT MODE)
4736 4545 5350 JMP ,+3 /START BY WAITING FOR TRANSFER REQUEST
4737 4546 1166 TAD GOOD
4738 4547 4442 XDROUT /TRANSFER OUT TO SECTOR BUFFER
4739 4550 4443 STR
4740 4551 7410 SKP
4741 4552 5346 JMP ,+4 /WAIT FOR TRANSFER REQUEST FLAG
4742 4553 4445 SDN
4743 4554 5350 JMP ,+4 /WAIT FOR DONE FLAG
4744 4555 4444 SER
4745 4556 4445 DK
4746 4557 4452 E210, ERROR /UNEXPECTED ERROR FLAG
4747 4560 4502 SCOPE
4748 4561 5741 JMP I FB128BYTES
4749
4750
4751
4752
4753
4754 4577 0002 PAGE
4755 4600 /INITIALIZE THE NUMBER OF SECTORS AVAILABLE TO ACCESS (PER TRACK) VIA THE

```

```

4756 /DIFFERENCE BETWEEN THE CONTENTS OF PROGRAM LOCATIONS " FIRST " AND " LAST ".
4757 /
4758 4600 4600 XINITSECTORS, .
4759 4601 1032 TAD LAST
4760 4602 7040 CHA
4761 4603 1031 TAD FIRST
4762 4604 3122 DCA SECTORS
4763 4605 3311 DCA XSTARGET
4764 4606 5600 JMP I XINITSECTORS
4765 /
4766 /GET A SECTOR
4767 /
4768 /SECTOR ACCESS 1=32 (OCTAL)
4769 /
4770 4607 4607 XGETASECTOR, .
4771 /
4772 /SET (SSTART)
4773 /
4774 4610 7301 XXGETASECTOR, CLL CLA IAC / 1
4775 4611 3310 DCA IF
4776 4612 1120 TAD RDC
4777 4613 7450 SNA
4778 4614 2310 ISZ IF / + 1 RDC
4779 4615 7710 SPA CLA
4780 4616 5222 JMP ,+4 / + 1 WRITE
4781 4617 7307 CLL CLA IAC RIL
4782 4620 0132 AND TESTP
4783 4621 7640 SZA CLA
4784 4622 2310 ISZ IF / + 1 8-BIT MODE
4785 4623 4777 JMS VT78CK /GO CHECK IF ON VT78 SYSTEM
4786 4624 2310 ISZ IF /INCREASE (IF) FOR VT78
4787 /
4788 / (IF) IS:
4789 /
4790 / 1 IF READ
4791 / 2 IF WRITE
4792 / 2 IF WRITE 8-BIT MODE
4793 / 2 IF READ AND PROGRAM VERIFY
4794 / 3 IF READ AND PROGRAM VERIFY 8-BIT MODE
4795 / 2 IF READ 8-BIT MODE
4796 /
4797 4625 1311 NEXTSECTOR, TAD XSTARGET
4798 4626 7440 SZA
4799 4627 1310 TAD IF /INTERLEAVE FACTOR 1, 2, 3
4800 4630 7001 IAC /NOW INTERLEAVE FACTOR IS 2, 3, 4
4801 4631 3311 DCA XSTARGET
4802 4632 1311 TAD XSTARGET
4803 4633 1376 TAD (-33)
4804 4634 7510 SPA
4805 4635 5243 JMP OKSTARGET / < 33
4806 4636 7640 SZA CLA /SKIP IF = 33
4807 4637 7240 STA
4808 4640 1375 TAD (-31)
4809 4641 1311 TAD XSTARGET
4810 4642 3311 DCA XSTARGET

```

```

4811 4643 7200 OKSTARGET, CLA
4812 4644 1311 TAD XSTARGET
4813 4645 3124 DCA STARGET
4814 /
4815 /PREVIOUSLY AT L/S THE FOLLOWING CONDITION WAS TRUE
4816 /
4817 / O < (FIRST) <= (LAST)
4818 /
4819 /THEREFORE TEST FOR THE CONDITION
4820 /
4821 / (STARGET) <= (LAST)
4822 /
4823 4646 1032 TAD LAST
4824 4647 7041 CIA
4825 4650 1124 TAD STARGET
4826 4651 7740 SMA SZA CLA
4827 4652 5225 JMP NEXTSECTOR
4828 /
4829 /AND FOR THE CONDITION
4830 /
4831 / (FIRST) <= (STARGET)
4832 /
4833 4653 1124 TAD STARGET
4834 4654 7041 CIA
4835 4655 1031 TAD FIRST
4836 4656 7740 SMA SZA CLA
4837 4657 5225 JMP NEXTSECTOR
4838 /FORMAT (XTARGET) BITS 0-6 TRACK ; AND BITS 7-11 SECTOR
4839 /
4840 4660 1131 TAD TARGET
4841 4661 7104 CLL RAL
4842 4662 7006 RTL
4843 4663 7006 PTL
4844 4664 1124 TAD STARGET
4845 4665 3151 DCA XTARGET
4846 /AND SET THE HEADER WORDS 0, AND 1 OF THE WRITE BUFFER FOR RDC I,D.
4847 /
4848 /WORD 0 IS THE TRACK
4849 /WORD 1 IS THE UNIT (BIT 4) AND THE SECTOR (BITS 7-11)
4850 /
4851 4666 1131 TAD TARGET
4852 4667 3774 DCA WBUFFER
4853 4670 1773 TAD UNITZ
4854 4671 7106 CLL RTL
4855 4672 7006 RTL
4856 4673 7006 RTL
4857 4674 7004 RAL

4858 4675 1124 TAD STARGET1
4859 4676 3772 DCA WBUFFER+1
4860 /WORDS 63 AND 64 OR 127 AND 128 ARE OUT-SUMCHECK WORDS
4861 /
4862 4677 1155 TAD PATSUMCHECK
4863 4700 1774 TAD WBUFFER
4864 4701 1772 TAD WBUFFER+1
4865 4702 3556 DCA I WORDX
    
```

```

4866 4703 1556 TAD I WORDX
4867 4704 1556 TAD I WORDX
4868 4705 7041 CIA
4869 4706 3557 DCA I WORDY
4870 4707 5607 JMP I XGETASECTOR
4871 /
4872 4710 0000 IF, 0
4873 4711 0000 XSTARGET, 0 / 1 TO 32
4874 /
4875 /RANDOM NUMBER GENERATOR
4876 / (EXIT IS WITH THE RANDOM # IN THE ACCUMULATOR)
4877 4712 4712 RANGEN,
4878 4713 7301 CLL CLA IAC
4879 4714 1326 TAD R1
4880 4715 1327 TAD R2
4881 4716 7106 CLL RTL
4882 4717 3326 DCA R1
4883 4720 1327 TAD R2
4884 4721 7012 RTR
4885 4722 1326 IAD R1
4886 4723 3327 DCA R2
4887 4724 1327 TAD R2
4888 4725 5712 JMP I RANGEN
4889 4726 1234 R1, 1234
4890 4727 0765 R2, 0765
4891 /
4892 /TESTS
4893 /
4894 4730 0600 TESTS, T0
4895 4731 0626 T1
4896 4732 0652 T2
4897 4733 0723 T3
4898 4734 1000 T4
4899 4735 1071 T5
4900 4736 1132 T6
4901 4737 1200 T7
4902 4740 1215 T10
4903 4741 1233 T11
4904 4742 1260 T12
4905 4743 1456 T13
4906 4744 1603 T14
4907 4745 1601 T15
4908 4746 1455 T16
4909 4747 1602 T17
4910 4750 1600 T20
4911 4751 1675 T21
4912 4752 1674 T22
4913 4753 2265 T23
4914 4754 1714 T24
4915 4755 1716 T25
4916 4756 1720 T26
4917 4757 2107 T27
4918 4760 2200 T30
4919 4761 2202 T31
4920 4762 2223 T32
    
```

```

4921 4763 2241 T33
4922 4764 2300 T34
4923 4765 2276 T35
4924 4766 2277 T36
4925 4767 2275 T37
4926 4770 0462 NOMORE=3
4927 4772 7115
4928 4773 4240
4929 4774 7114
4930 4775 7747
4931 4776 7745
4932 4777 1165
4933 5000
4934 5001
4935 5002
4936 5000 5000 XTY4OCT,
4937 5001 7410 SKP
4938 5002 3162 DCA TTYBUSY /INITIALIZE TTYBUSY INDICAOR.
4939 5003 4500 SETUP
4940 5004 5000 XTY4OCT
4941 5005 1600 TAD I XTY4OCT
4942 5006 2200 ISZ XTY4OCT
4943 5007 3262 DCA XOCTAL /FOR " OCTAL " ADDRESS
4944 5010 1662 TAD I XOCTAL
4945 5011 3262 DCA XOCTAL /OCTAL
4946 5012 7346 CLL STA RTL / -3
4947 5013 1117 TAD K7777 / -1
4948 5014 3263 DCA DIGITS
4949 5015 7346 SHIFT, CLL STA RTL / -3
4950 5016 3265 DCA SHIFTS
4951 5017 1262 TAD XOCTAL
4952 5020 7100 XSHIFT, CLL
4953 5021 7510 SPA
4954 5022 7020 CML
4955 5023 7004 RAL
4956 5024 2265 ISZ SHIFTS
4957 5025 5220 JMP XSHIFT
4958 5026 3262 DCA XOCTAL /NEW
4959 5027 1262 TAD XOCTAL
4960 5030 0174 AND K0007 /OCTAL MASK
4961 5031 7450 SVA
4962 5032 5242 JMPD, JMP ISZDIG /DO NOT PRINT LEADING ZEROS.
4963 5033 3264 DCA XXDMP /STORE NUMBER TO BE PRINTED
4964 5034 1173 TAD K7000
4965 5035 3232 DCA -3 /NOP JMP ISZDIG, LEADING ZEROS WILL NOT BE PRINTED
4966 5036 1264 TAD XXDMP /RETURN VALUE TO BE PRINTED.
4967 5037 1377 TAD (260) /FOR ASCII COMPONENT
4968 5040 4506 TYPEIT
4969 5041 2266 ISZ CCNT /INDICATES A CHARACTER HAS BEEN PRINTED
4970 5042 2263 ISZDIG, ISZ DIGITS /INDEX DIGIT COUNT
4971 5043 5215 JMP SHIFT
4972 5044 2175 ISZ XCNT /TY8OCT?
4973 5045 5202 JMP XTY4OCT+2 /YES
4974 5046 1266 TAD CCNT

```

```

4975 5047 7650 SVA CLA /PRINT ONLY A ZERO??
4976 5050 5257 JMP ZERO /YES
4977 5051 7340 CLL CLA CMA
4978 5052 3175 DCA XCNT /INIT COUNTER
4979 5053 3266 DCA CCNT
4980 5054 1376 TAD (JMP ISZDIG) /ESTABLISH FIRST TIME SWITCH.
4981 5055 3232 DCA JMPDIG
4982 5056 5775 JMP THENEXIT
4983 5057 1377 ZERO, TAD (260)
4984 5060 4506 TYPEIT
4985 5061 5251 JMP -10
4986 5062 0000 XOCTAL, 0 / " OCTAL " FOR TYPEDOUT
4987 5063 7775 DIGITS, -3
4988 5064 0000 XXDMP, 0
4989 5065 7775 SHIFTS, -3
4990 5066 0000 CCNT, 0
4991 /THIS ROUTINE WILL DETERMINE DEVICE CODE TO USE IN PLACE
4992 /OF THE STANDARD 75 CODE.
4993 /
4994 /*****
4995 CHNDEV, 0
4996 TAD DTESTP /GET TESTING PARAMETERS.
4997 AND (700) /ISOLATE DEVICE CODE TO USE.
4998 SVA /USE 75??
4999 JMP I CHNDEV /YES.
5000 CLL RTR
5001 RAR /MOVE TO BIT POSITION 6-8.
5002 TAD (6700)
5003 /PROPER DEVICE CODE SHOULD NOW BE ESTABLISHED.
5004 /IF THE OPERATOR HAS MADE AN ERROR THE PROGRAM HAS NO WAY
5005 /OF KNOWING IT.
5006 /
5007 DCA A10 /SET DEVICE CODE = 67X-
5008 TAD (XDEVIC-1)
5009 DCA A11 /INTO APPLICABLE PROGRAM
5010 TAD I A11
5011 SVA
5012 JMP +6 /DEVICE CODE LOCATIONS
5013 DCA GOOD
5014 TAD A10
5015 DCA I GOOD /SPECIFIED AT
5016 ISZ A10
5017 JMP -7 /PROGRAM LOCATION "DEVICE"
5018 TAD I XK67X2A
5019 DCA K67X2B /DUPLICATE IOT CODE 67X2
5020 TAD I XK67X2A
5021 DCA K67X2C /DUPLICATE IOT CODE 67X2
5022 TAD I XK67X3A
5023 DCA K67X3B /DUPLICATE IOT CODE 67X3
5024 TAD I XK67X4A
5025 DCA K67X4B /DUPLICATE IOT CODE 67X4
5026 TAD I XK67X5A
5027 DCA K67X5B /DUPLICATE IOT CODE 67X5
5028 TAD I XK67X6A
5029 DCA K67X6B /DUPLICATE IOT CODE 67X6

```

```

5030 5126 1741 TAD I XK67X6A
5031 5127 3763 DCA K67X6C /DUPLICATE IOT CODE 67X6
5032 5130 1742 TAD I XK67X7A
5033 5131 3762 DCA K67X7B /DUPLICATE IOT CODE 67X7
5034 5132 5667 JMP I CRNDEV /EXIT AND RUN PROGRAM.
5035 /PROGRAM DEVICE CODE TABLE
5036
5037 5133 6201 XDEVICE, K67X0 /VT78/SEL = SELECT UNIT A OR UNIT B
5038 5134 6206 K67X1
5039 5135 6401 XK67X2A, K67X2A /XDR TRANSFER DATA REGISTER
5040 5136 6410 XK67X3A, K67X3A / STR (SKIP ON TRANSFER REQUEST FLAG)
5041 5137 6415 XK67X4A, K67X4A / SER (SKIP ON ERROR FLAG)
5042 5140 6422 XK67X5A, K67X5A / SDN (SKIP ON DONE FLAG)
5043 5141 6431 XK67X6A, K67X6 / CLEAR (AC = 0) / SET (AC = 1) INTERRUPT ENABLE
5044 5142 6435 XK67X7A, K67X7A / INIT (INITIALIZE / IMPLIED READ TRACK 0 SECTOR 1)
5045 5143 0000 0
5046 /
5047 /
5048 /
5049 /
5050 /
5051 /
5052 /ROUTINE TO DETERMINE IF ON APT.
5053
5054 5144 0000 XCHK22, 0
5055 5145 1022 TAD 22
5056 5146 0361 AND (4000
5057 5147 7640 SZA CLA
5058 5150 5744 JMP I XCHK22
5059 5151 2344 ISZ XCHK22
5060 5152 5744 JMP I XCHK22
5061
5062 5161 4000
5063 5162 6451
5064 5163 3637
5065 5164 3622
5066 5165 0701
5067 5166 0715
5068 5167 0707
5069 5170 3623
5070 5171 6404
5071 5172 5132
5072 5173 6700
5073 5174 0700
5074 5175 4114
5075 5176 5242
5076 5177 0260
5077 5200 PAGE
5078 /
5079 /ROUTINE TO TYPE 8 OCTAL DIGITS.
5080 /FORMAT TY8OCT; MSB(STARTING ADDRESS OF DOUBLE LOG TO BE PRINTED).
5081 XTY8OCT, 0
5082 5201 1600 TAD I XTY8OCT /GET STARTING ADDRESS
5083 5202 3212 DCA ,+10

```

```

5084 5203 1600 TAD I XTY8OCT
5085 5204 7001 IAC /SECOND ADDRESS
5086 5205 3211 DCA ,+4
5087 5206 7344 CIL CLA CMA RAL /-2
5088 5207 3175 DCA XCNT
5089 5210 4505 TY4OCT
5090 5211 7000 NOP /REPLACED WITH POINTER TO BE
5091 5212 7000 NOP /PRINTED
5092 5213 2200 ISZ XTY8OCT /UPDATE RETURN
5093 5214 5600 JMP I XTY8OCT /EXIT
5094 /SUBROUTINE ; TAB
5095 /ENTRY ; TAB; +N
5096 /COMMENT ; PRINT " N " SPACES WHERE N IS VIA INDEXED XTAB
5097 /
5098 5215 5215 XTAB, .
5099 5216 4500 SETUP
5100 5217 5215 XTAB
5101 5220 3243 DCA XXTAB /FOR COUNT
5102 5221 7200 XTABL, CLA
5103 5222 1243 TAD XXTAB
5104 5223 1615 TAD I XTAB /FOR " N "
5105 5224 3243 DCA XXTAB
5106 5225 1320 TAD CHARLINE / # OF CHARACTERS ALREADY TYPED ON THIS LINE
5107 5226 1377 TAD (110) / 72 CHARACTER LINE STANDARD
5108 5227 7041 CIA
5109 5230 1243 TAD XXTAB
5110 5231 7550 SPA SNA
5111 5232 5221 JMP XTABL
5112 5233 7041 CIA
5113 5234 3243 DCA XXTAB
5114 5235 2215 ISZ XTAB
5115 5236 1376 TAD (240)
5116 5237 4506 TYPEIT
5117 5240 2243 ISZ XXTAB
5118 5241 5236 JMP ,+3
5119 5242 5775 JMP THENEXIT
5120 5243 0000 XXTAB, 0
5121 /
5122 /SUBROUTINE ; PRINT
5123 /ENTRY ;
5124 /COMMENT ; PRINT A " MESSAGE ", AND A <CR><LF> AT EACH _
5125 /
5126 / CALL SYNTAX FOR PRINT
5127 /
5128 / 1. PRINT; MTEXT
5129 /
5130 5244 5244 XPRINT, .
5131 5245 4472 CHEK22
5132 5246 5341 JMP PNTXT
5133 5247 4507 WAITTY
5134 5250 1644 TAD I XPRINT / " TEXT"
5135 5251 2244 ISZ XPRINT
5136 5252 3262 DCA MESSAGE /ADDRESS
5137 5253 1662 NUWORD, TAD I MESSAGE
5138 5254 4263 OUTPUT /LEFT BYTE

```

```

5139 5255 1662 TAD I MESSAGE
5140 5256 2262 ISZ MESSAGE
5141 5257 4433 BSW
5142 5260 4263 OUTPUT /RIGHT BYTE
5143 5261 5253 JMP NUWORD
5144 5262 0000 MESSAGE,0
5145 5263 4263 OUTPUT=JMS
5146 5263 5263 XOUTPUT,
5147 5264 0374 AND (-100) /MASK MS BITS 0-5
5148 5265 7440 SZA
5149 5266 5271 JMP ,+3
5150 5267 3162 DCA TTYBUSY
5151 5270 5773 JMP PLEXIT
5152 5271 1372 TAD (4100)
5153 5272 7450 SNA
5154 5273 5307 JMP NULINE /
5155 5274 1371 TAD (-4100+2) /CODE 200
5156 5275 7500 SNA
5157 5276 7001 IAC /CODE 300
5158 5277 4433 BSW
5159 5300 4506 TYPEIT
5160 5301 5663 JMP I XOUTPUT
5161 5302 1263 NULINE, TAD XOUTPUT
5162 5303 3305 DCA XTYPEIT
5163 5304 5311 JMP XNULINE
5164 5305 5305 XTYPEIT,
5165 5306 4321 JMS TYIASC
5166 5307 2320 ISZ CHARLINE
5167 5310 5705 JMP I XTYPEIT
5168 5311 1370 XNULINE,TAD (15) / <CR>
5169 5312 4321 JMS TYIASC
5170 5313 1367 TAD (12) / <LF>
5171 5314 4321 JMS TYIASC
5172 5315 1366 TAD (-110)
5173 5316 3320 DCA CHARLINE / 72 CHARACTER LINE (NEGATIVE NOTATION)
5174 5317 5705 JMP I XTYPEIT
5175 5320 7670 CHARLINE, -110 / # CHARACTERS PER LINE ARE COUNTED HERE
5176 /
5177 /
5178 /
5179 5321 5321 TYIASC,
5180 5322 6002 IOF
5181 5323 3014 DCA A14
5182 5324 4472 CHEK22
5183 5325 5341 JMP PNTXT
5184 5326 4765 JMS CKSFLG /GO CHECK IF "S IS ACTIVE
5185 5327 1014 TAD A14

5186 5330 6046 ILS
5187 5331 7200 CLA
5188 5332 1162 TAD TTYBUSY
5189 5333 7640 SZA CLA
5190 5334 5773 JMP PLEXIT
5191 5335 2162 ISZ TTYBUSY
5192 5336 1364 TAD (PI)
5193 5337 3002 DCA IPI

```

```

5194 5340 7410 SKP
5195 5341 2244 PNTXT, XPRINT /UPDATE RETURN,
5196 5342 6001 ISZ
5197 5343 5644 JMW I XPRINT
5198 /
5199 /
5200 5344 6042 XTCF, TCF
5201 5345 1014 TAD A14
5202 5346 1363 TAD (-207)
5203 5347 7650 SNA CLA
5204 5350 5705 JMP I XTYPEIT
5205 5351 5721 JMP I TYIASC
5206 /
5207 /
5208 /
5209 5352 5352 XSPECIALTYPEIT,
5210 5353 4500 SETUP
5211 5354 5352 XSPECIALTYPEIT
5212 5355 1752 TAD I XSPECIALTYPEIT
5213 5356 2352 ISZ XSPECIALTYPEIT
5214 5357 4506 TYPEIT
5215 5360 5775 JMP THENEXIT
5216 5363 7571
5217 5364 5417
5218 5365 5400
5219 5366 7670
5220 5367 0012
5221 5370 0015
5222 5371 3702
5223 5372 4100
5224 5373 5510
5225 5374 7700
5226 5375 4114
5227 5376 0240
5228 5377 0110
5229 5400 PAGE
5230 /ROUTINE TO CHECK IF "S IS ACTIVE
5231 5400 0000 CKSFLG, 0
5232 5401 7001 IAC /AC=1
5233 5402 6211 CDF 10
5234 5403 1777 TAD C8SFLG /GO GET FLAG (7777 IF SET)
5235 5404 6201 CDF 0
5236 5405 7640 SZA CLA /IS FLAG SET?
5237 5406 5600 JMP I CKSFLG /NO - RETURN
5238 5407 6212 CIF 10 /CHANGE INSTR FIELD TO CONSOLE PACKAGE
5239 5410 4776 JMS CBENTR /GO TO CONSOLE PACKAGE
5240 5411 5600 JMP I CKSFLG /RETURN
5241 /
5242 /
5243 /
5244 /
5245 5412 5412 XWAITTY,
5246 5413 1162 TAD TTYBUSY
5247 5414 7640 SZA CLA

```

```

5248 5415 5213      JMP ,=2
5249 5416 5612      JMP I XWALTTY
5250                /ENTRY TO THIS POINT WAS CAUSED BY A PROGRAM INTERRUPT REQUEST
5251                /
5252 5417 3315      PI,      DCA XAC
5253 5420 7010      RAR
5254 5421 3316      DCA XLINK      /SAVE (AC) AND (LINK)
5255 5422 6031      KSF
5256 5423 7410      SKP
5257 5424 5775*     JMP XKCC      /IGNORE KEYBOARD IRQ
5258 5425 6041      TSF
5259 5426 5234      JMP PISON
5260                /IF THIS TELEPRINTER FLAG IS EXPECTED (TIYBUSY) = 1
5261                /THEN "JMP XICF"
5262                /IF NOT THEN "JMP PIEEXIT"
5263                /
5264 5427 1162      TAD TIYBUSY
5265 5430 7740      SNA SZA CLA
5266 5431 5774*     JMP XICF
5267 5432 6042      TCF
5268 5433 5310      JMP PIEEXIT
5269                /IF (BUSY) = 1, THEN AN RX01 PI IS ALREADY BEING PROCESSED
5270                /
5271                /IF (GOBIT) = 0, THEN THIS DISKETTE IRQ IS UNEXPECTED
5272                /
5273                /DISABLE RX01 INTERRUPT
5274                /
5275                /READ RX01 STATUS REGISTER
5276                /
5277 5434 1162      PISON, TAD TIYBUSY
5278 5435 1111      TAD BUSY
5279 5436 7640      SZA CLA
5280 5437 5310      JMP PIEEXIT
5281 5440 2111      ISZ BUSY
5282                /REFRESH PROGRAM LOCATION " FORCE "
5283                /
5284 5441 3325      DCA XFORCE
5285                /
5286 5442 4445      SDN
5287 5443 5305      JMP UNKNOWN
5288 5444 1154      TAD GOBIT
5289 5445 7750      SPA SNA CLA
5290 5446 5773*     JMP SONUNEXPECTED
5291 5447 4446      INIR
5292 5450 4441      XDRIN
5293 5451 0372      AND (377)
5294 5452 3126      DCA ASTATUS

5295                /IF THIS IS TEST # 30 THEN IGNORE DELETED DATA MARKS (IF ANY)
5296                /
5297 5453 1171      TAD TEST
5298 5454 1371      TAD (-T30)
5299 5455 7650      SNA CLA
5300 5456 5275      JMP DDIGNORE
5301                /TECHNICAL NOTE:
5302                /

```

```

5303                /THE COMMANDS "FILL BUFFER" (0), AND "EMPTY BUFFER" (2)
5304                /NEVER SHOULD ATTRACT THE "DELETED DATA" STATUS (100)
5305                /
5306 5457 1370      TAD (16)      /COMMAND MASK
5307 5460 0112      AND COMMAND      /FB (0), OR EB (2)
5308 5461 7440      SZA      /SKIP IF FILL BUFFER COMMAND (0)
5309 5462 7112      CLL RTR
5310 5463 7640      SZA CLA      /SKIP IF EMPTY BUFFER COMMAND (2)
5311 5464 7305      CLL CLA IAC RAL      / 2
5312 5465 0132      AND TESTP
5313 5466 7112      CLL RTR      /PUT TO LINK
5314 5467 1367      TAD (100)      / 100
5315 5470 0126      AND ASTATUS      / A STATUS D,D. MASK
5316                /
5317                /IF (L) = 0 AND (AC) = 0, O,K, - NO D,D. MARK
5318                /
5319                /IF (L) = 0 AND (AC) > 0 (=100), UNEXPECTED D,D.
5320                /
5321                /IF (L) = 1 AND (AC) = 0, D,D. MARK EXPECTED DIDN'T OCCUR
5322                /
5323                /IF (L) = 1 AND (AC) > 0 (=100), O,K, - D,D. MARK OCCURED
5324                /
5325 5471 7430      SZL
5326 5472 7640      SZA CLA      / (L) = 1
5327 5473 7440      SZA      / (L) = 0, OR (L) = 1 AND (AC) > 0
5328 5474 5337      JMP DDERROR      / (L) = 0 AND (AC) > 0 (=100) , OR (L) = 1 AND (AC) = 0
5329 5475 4444      DDIGNORE, SER
5330 5476 5300      JMP VERIFY      /RX01 OK - RETURN TO INLINE CODE
5331 5477 5771*     JMP RXERROR
5332                /
5333                /VERIFY THAT THE CONTENTS OF THE A-STATUS REGISTER = 0
5334                /
5335                /WHEN NO RX01 ERROR FLAG EXISTS
5336                /
5337                /MASK BITS 4 (DRIVE READY); AND 5 (DELETED DATA)
5338                /
5339 5500 1126      VERIFY, TAD ASTATUS
5340 5501 0366      AND (73)
5341 5502 7640      SZA CLA
5342 5503 5765*     JMP NOSER
5343 5504 5321      JMP XRETURN
5344                /
5345                /AN UNKNOWN PROGRAM INTERRUPT OCCURED
5346                /
5347 5505 4473      UNKNOWN, PRINT
5348 5506 7070      MUNKNOWN
5349 5507 3111      DCA BUSY
5350                /
5351 5510 1316      PIEEXIT, TAD XLINK
5352 5511 7104      CLL RAL
5353 5512 1315      TAD XAC
5354 5513 6001      ION
5355 5514 5400      JMP I 0
5356 5515 0000      XAC, 0
5357 5516 0000      XLINK, 0

```



```

5358 /THE CONTENTS OF RETURN ARE SETUP WITHIN THE SUBROUTINES "LCD-A" AND "LCD-B"
5359 /TO REPRESENT THE RETURN ADDRESS OF THE INLINE TESTING
5360 /
5361 5517 0000 RETURN, 0
5362 /
5363 5520 2317 ERETURN, ISZ RETURN /INCREMENT FOR ERROR RETURN ADDRESS
5364 /
5365 /ENTRY TO HERE FROM PI SERVICE
5366 /
5367 /NO RX01 ERROR FLAG EXISTS
5368 /
5369 5521 3111 XRETURN, DCA BUSY
5370 5522 4466 TICK /TIMING FOR APT IF NEEDED.
5371 5523 6001 ION
5372 5524 5717 JMP I RETURN
5373 /...ENTRY TO THIS POINT MAY HAVE BEEN FROM WITHIN THE SUBROUTINE "COMPARE"
5374 /WHICH DETECTED A DATA NO STATUS ERROR (DNS), OR
5375 /...ENTRY TO THIS POINT MAY HAVE BEEN FROM ROUTINES "XHUNGUP" OR "HUNGUP"
5376 /THEREBY FORCING AN ERROR INFORMATION PRINTOUT
5377 /
5378 5525 0000 XXFORCE, 0
5379 5526 2111 ISZ BUSY
5380 5527 5771 JMP RXERROR
5381 5530 7240 XFORCE, STA
5382 5531 1111 TAD BUSY
5383 5532 3111 DCA BUSY
5384 5533 6001 ION
5385 5534 5725 JMP I XXFORCE
5386 5535 7020 DTYPE, MEDDDIDNOT
5387 5536 7031 MUDDID
5388 /A DISKETTE DELETED DATA MALFUNCTION HAS BEEN DETECTED
5389 /
5390 / IF (AC) = 0 - EXPECTED D.D. DIDN'T OCCUR
5391 / IF (AC) = 100 - UNEXPECTED D.D. OCCURED
5392 /
5393 5537 7640 DDERROR, SZA CLA / 0 OR 1
5394 5540 7001 IAC
5395 5541 1364 TAD (DTYPE)
5396 5542 3137 DCA XXX
5397 /PROGRAMMING NOTE: "SER" TO CLEAR ACCOMPANYING ERROR FLAG (IF ANY)
5398 /
5399 5543 4444 SER
5400 5544 7000 NOP
5401 5545 1537 TAD I XXX
5402 5546 5771 JMP RXERROR
5403 5564 5535
5404 5565 3322

5405 5566 0073
5406 5567 0100
5407 5570 0016
5408 5571 5600
5409 5572 0377
5410 5573 3320
5411 5574 5344
5412 5575 0323
    
```

```

5413 5576 0200
5414 5577 0404
5415 5600
5416 PAGE
5417 /A DISKETTE ERROR HAS BEEN DETECTED
5418 /
5419 / (DMTYPE) NOT = 0 IF A D.D. ERROR EXISTS
5420 / (DMTYPE) = 0 IF NO D.D. ERROR EXISTS
5421 /
5422 5600 3777 RXERROR, DCA DMTYPE
5423 /
5424 5601 1112 TAD COMMAND
5425 5602 3152 DCA ECOMMAND
5426 5603 1776 TAD XXFORCE
5427 5604 1375 TAD (-XHUNG)
5428 5605 7650 SNA CLA
5429 5606 5246 JMP EERROR
5430 5607 1776 TAD XXFORCE
5431 5610 1374 TAD (-XCOMPARE)
5432 5611 7650 SNA CLA
5433 5612 5246 JMP EERROR
5434 5613 1373 SAVEBSTATUS, TAD (16)
5435 5614 4436 LCD
5436 5615 4471 WAIT
5437 5616 4445 SDN
5438 5617 5215 JMP , -2
5439 5620 3006 DCA 6 /WAIT POINTER
5440 5621 4444 SER
5441 5622 7000 NOP
5442 5623 4441 XDRIN
5443 5624 0372 AND (377)
5444 5625 3127 DCA BSTATUS
5445 5626 1771 SAVECSTATUS, TAD UNIT
5446 5627 1370 TAD (12)
5447 5630 4436 LCD
5448 5631 4471 WAIT
5449 5632 4445 SDN
5450 5633 5231 JMP , -2
5451 5634 3006 DCA 6 /WAIT POINTER
5452 5635 4444 SER
5453 5636 7000 NOP
5454 5637 4441 XDRIN
5455 5640 0372 AND (377)
5456 5641 3130 DCA CSTATUS
5457 /PRINT AN ERROR MESSAGE IF AC SW 4 = 0
5458 /
5459 5642 4572 LASSW4, LAS
5460 5643 0367 AND (SW4)
5461 5644 7640 SZA CLA
5462 5645 5766 JMP NOPRINT
5463 /THE FOLLOWING INFORMATION IS PRINTED FOR ALL ERRORS DETECTED
5464 /
5465 /THE ERROR HEADER TEXT IS INHIBITED IF THE ERROR IS NOT THE FIRST ERROR EVER
5466 /
    
```

```

5467 / FAT CMND XDR CODE RST START TARGET XXXX PASS
5468 /
5469 /
5470 /
5471 5646 1171 ERROR, TAD TEST /GET TEST IN ERROR
5472 5647 4467 AERROR /REPORT ERROR TO APT.
5473 5650 2115 ISZ FIRSTERROR
5474 5651 5255 JMP NOHEADER
5475 5652 4473 PRINT
5476 5653 6627 MEHEADER
5477 5654 5271 JMP ONECRLF
5478 /IF THIS IS =NOT= A FORCED TYPEOUT, AND IF THERE ARE NO DATA COMPARE
5479 /ERRORS (COMPRERRUR=0), THEN PRINT ONLY 1-CRLF
5480 /BECAUSE
5481 /
5482 /THIS ERROR MUST BE AN ERROR AT THE END OF THE EMPTY BUFFER DONE FLAG
5483 /WHICH WOULD BE ASSOCIATED TO ANY PREVIOUS FORCED TYPEOUT OF DATA ERRORS
5484 /
5485 5655 1776* NOHEADER, TAD XXFORCE
5486 5656 7650 SNA CLA
5487 5657 5267 JMP TWOCRLF
5488 5660 1113 TAD COMPRERRUR
5489 5661 7640 SZA CLA
5490 5662 5271 JMP ONECRLF
5491 /IF THIS IS A FORCED TYPEOUT FROM " XXINIT " THEN PRINT 1 CRLF
5492 /
5493 5663 1776* TAD XXFORCE
5494 5664 1365 TAD (-XXINIT)
5495 5665 7650 SNA CLA
5496 5666 5271 JMP ONECRLF
5497 5667 4473 TWOCRLF, PRINT
5498 5670 6602 MCRLF
5499 5671 4473 ONECRLF, PRINT
5500 5672 6602 MCRLF
5501 5673 4505 TY4OCT
5502 5674 0171 TEST
5503 5675 4504 TAB
5504 5676 0005 5
5505 5677 4764* JMS INITSWITCH
5506 5700 5305 JMP ,+5
5507 /IF AN ERROR FROM THE RECAL THEN PRINT [INIT] FOR THE COMAND
5508 /
5509 5701 1776* TAD XXFORCE
5510 5702 1365 TAD (-XXINIT)
5511 5703 7640 SZA CLA
5512 5704 5310 JMP ,+4
5513 5705 4473 PRINT
5514 5706 6671 MINIT
5515 5707 5312 JMP TAB12
5516 5710 4505 TY4OCT
5517 5711 0152 ECOMMAND
5518 5712 4504 TAB12, TAB
5519 5713 0012 12
5520 /
5521 /IF THE DEVICE TEST IS HUNG, THEN THE A-, B-, AND C- STATUS IS NOT APPLICABLE

```

```

5522 /
5523 5714 1776* TAD XXFORCE
5524 5715 1375 TAD (-XHUNG)
5525 5716 7650 SNA CLA
5526 5717 5325 JMP DASHALL
5527 /IF THIS IS A " FORCED " TYPEOUT THEN THE B- AND C-STATUS REGISTERS
5528 /ARE NOT APPLICABLE TO THIS TYPEOUT BECAUSE THEY ARE RESIDUAL FROM THE
5529 /PREVIOUS COMMAND WHICH WOULD HAVE HAD A PREVIOUS ERROR TYPE OUT
5530 /RELATING TO THE B- AND C-STATUS REGISTERS IF AN ERROR HAD OCCURED
5531 /
5532 /A DATA NO ERROR STATUS HAS BEEN DETECTED PRIOR TO THE COMPLETION OF
5533 /THE EMPTY BUFFER FUNCTION
5534 /
5535 5720 1776* TAD XXFORCE
5536 5721 1374 TAD (-XCOMPARE)
5537 5722 7640 SZA CLA
5538 5723 5340 JMP TYASTATUS
5539 5724 5331 JMP DASHBC
5540 5725 4473 DASHALL, PRINT
5541 5726 7017 MDASH
5542 5727 4504 TAB
5543 5730 0017 17
5544 5731 4473 DASHBC, PRINT
5545 5732 7017 MDASH
5546 5733 4504 TAB
5547 5734 0024 24
5548 5735 4473 PRINT
5549 5736 7017 MDASH; JMP TAB31
5550 5737 5763* TYASTATUS, TY4OCT
5551 5740 4505 ASTATUS
5552 5741 0126 TAB
5553 5742 4504 17
5554 5743 0017 17
5555 5744 4505 TY4OCT
5556 5745 0127 BSTATUS
5557 5746 4504 TAB
5558 5747 0024 24
5559 5750 5762* JMP TYCSTATUS
5560 5762 6000
5561 5763 6002
5562 5764 6115
5563 5765 1332
5564 5766 6061
5565 5767 0200
5566 5770 0012
5567 5771 4242
5568 5772 0377
5569 5773 0016
5570 5774 4115
5571 5775 4434
5572 5776 5525
5573 5777 6060
5574 6000 PAGE
5575 6001 4505 TYCSTATUS, TY4OCT
5576 6001 0130 CSTATUS

```

```

5576 6002 4504 TAB31, TAB
5577 6003 0031 31
5578 6004 4315 JMS INITSWITCH
5579 6005 5211 JMP ,+4
5580 /IF (SSTART) = 0 THEN PRINT "HOME" BECAUSE A RECAL HAS TAKEN PLACE
5581 /THEREFORE THE ACTUATOR IS AT TRACK 0 (HOME)
5582 /
5583 6006 1123 TAD SSTART
5584 6007 7640 SZA CLA
5585 6010 5214 JMP ,+4
5586 6011 4473 PRINT
5587 6012 6665 MHOME
5588 6013 5222 JMP TAB43
5589 6014 4505 TY4OCT
5590 6015 0125 START
5591 6016 4504 TAB
5592 6017 0036 36
5593 6020 4505 TY4OCT
5594 6021 0123 SSTART
5595 6022 4504 TAB43, TAB
5596 6023 0043 43
5597 6024 4315 JMS INITSWITCH
5598 6025 5232 JMP PHUME
5599 /IF (FORCE) = THE ADDRESS OF "XXINIT" THEN ALSO PRINT [HOME] FOR THE TARGET
5600 /
5601 6026 1777* TAD XXFORCE
5602 6027 1376 TAD (-XXINIT)
5603 6030 7640 SZA CLA
5604 6031 5235 JMP ,+4
5605 6032 4473 PHOME, PRINT
5606 6033 6665 MHOME
5607 6034 5243 JMP ,+7
5608 6035 4505 TY4OCT
5609 6036 0131 TARGET
5610 6037 4504 TAB
5611 6040 0050 50
5612 6041 4505 TY4OCT
5613 6042 0124 STARGET
5614 6043 4504 TAB
5615 6044 0055 55
5616 6045 4505 TY4OCT
5617 6046 0132 TESTP
5618 6047 4504 TAB
5619 6050 0002 2
5620 6051 4434 TY8OCT
5621 6052 0163 PASS
5622 6053 0164 PASS+1

5623 6054 1269 TAD DMTYPE
5624 6055 7650 SNA CLA
5625 6056 5261 JMP ,+3
5626 6057 4473 PRINT
5627 6060 0000 DMTYPE, 0
5628 /
5629 /AC SW 11 TO INHIBIT RINGING OF BELL AT ERROR
5630 /

```

```

5631 6061 3115 NOPRINT, DCA FIRSTERROR
5632 6062 4572 LAS
5633 6063 0375 AND (SW11)
5634 6064 7640 SZA CLA
5635 6065 5270 JMP ,+3
5636 6066 4501 SPECIALTYPEIT
5637 6067 0207 BELL
5638 /IF ENTRI WAS FROM A "JMS FORCE" THEN EXIT BY A "JMP I FORCE"
5639 /
5640 6070 1777* TAD XXFORCE
5641 6071 7640 SZA CLA
5642 6072 5774* JMP XFORCE
5643 /
5644 /DEFINITIVE ERROR CODES AND MEANINGS
5645 /
5646 / 0 /NO ERROR
5647 / 10 /DRIVE 0 FAILED TO SEE HOME ON INITIALIZE
5648 / 20 /DRIVE 1 FAILED TO SEE HOME ON INITIALIZE
5649 / 30 /FOUND HOME WHEN STEPPING OUT 10 TRACKS FOR INIT
5650 / 40 /TRIED TO ACCESS A TRACK GREATER THAN 77
5651 / 50 /HOME WAS FOUND BEFORE DESIRED TRACK WAS REACHED
5652 / 60 /SELF DIAGNOSTIC ERROR
5653 / 70 /DESIRED SECTOR COULD NOT BE FOUND AFTER LOOKING AT 52 HEADERS
5654 / 100 /WRITE PROTECT ERROR
5655 / 110 /MORE THAN 40US AND NO SEP CLOCK SEEN
5656 / 120 /A PREAMBLE COULD NOT BE FOUND
5657 / 130 /PREAMBLE FOUND BUT NO ID MARK FOUND WITHIN ALLOWABLE TIME
5658 / 140 /HEADER CRC ERROR
5659 / 150 /THE HEADER TRACK ADDRESS OF A GOOD HEADER DOES NOT COMPARE
5660 / WITH THE DESIRED TRACK
5661 / 160 /TO MANY TRIES FOR A IDAM
5662 / 170 /DATA AM NOT FOUNT WITHIN ALLOTTED TIME
5663 / 200 /DATA CRC ERROR
5664 / 210 /ALL PARITY ERRORS
5665 /
5666 /RECAL IF DEFINITIVE ERROR CODE IS A SEEK ERROR
5667 /
5668 / (NOT CODES 140, 200, OR 210)
5669 /
5670 6073 4572 RECALIF, LAS
5671 6074 0373 AND (SW6)
5672 6075 7640 SZA CLA
5673 6076 5311 JMP LASSWO
5674 6077 1127 TAD BSTATUS
5675 6100 1372 TAD (-140)
5676 6101 7450 SNA
5677 6102 5311 JMP LASSWO
5678 6103 1371 TAD (-40)
5679 6104 7450 SNA
5680 6105 5311 JMP LASSWO
5681 6106 1370 TAD (-10)
5682 6107 7640 SZA CLA
5683 6110 4447 INIT
5684 6111 4572 LASSWO, LAS
5685 6112 7700 SNA CLA

```

```

5686 6113 4461 HLT16, HLT /AC SW 0 = 1 (HALT ON ERROR)
5687 6114 5767* JMP ERETURN
5688
5689 6115 6115 INITSWITCH, .
5690 6116 1766* TAD XWAIT
5691 6117 1365 TAD (-SDNSECOND)
5692 6120 7640 SZA CLA
5693 6121 2315 ISZ INITSWITCH
5694 6122 5715 JMP I INITSWITCH
5695
5696 /GENERATES TIMING FOR APT IF NEEDED.
5697
5698 6123 0000 XOK, 0
5699 6124 7000 NOP
5700 6125 2323 ISZ XOK /SKIP.
5701 6126 5723 JMP I XOK
5702
5703 /VT78/ROUTINE TO SELECT WHICH FLOPPY UNITS TO TEST
5704
5705 6127 0000 SELUNT, 0
5706 6130 1114 TAD DTESTP /GET TEST PAMETERS
5707 6131 0373 AND (40 /TEST BIT 6
5708 6132 7640 SZA CLA /DO ONE OR BOTH UNITS?
5709 6133 7001 IAC /BOTH
5710 6134 7040 CMA /EITHER A OR B
5711 6135 3351 DCA CHECKU /7777=EITHER , 7776=BOTH
5712 6136 1351 TAD CHECKU
5713 6137 7001 IAC
5714 6140 7640 SZA CLA
5715 6141 5346 JMP SETUNT /BOTH UNITS SELECTED- DO A THEN B
5716 6142 1114 TAD DTESTP
5717 6143 0364 AND (1000 /TEST BIT 2
5718 6144 7640 SZA CLA /A OR B?
5719 6145 7001 IAC /B ONLY
5720 6146 3135 SETUNT, DCA UNITCK /UNIT A=0 UNIT B=1
5721 6147 4451 CKUNIT /SET UP FOR UNIT A OR UNIT B
5722 6150 5727 JMP I SELUNT /RETURN
5723 CHECKU, 0
5724 6151 0000
5725 6165 1340
5726 6166 1722
5727 6167 5520
5728 6170 7770
5729 6171 7740
5730 6172 7640
5731 6173 0040
5732 6174 5530
5733 6175 0001
5734 6176 1332
5735 6177 5525
5736 PAGE
5737 /VT78/ SELECT IOT TO SELECT EITHER UNIT A OR UNIT B
5738 /IF AC11 (0) SELECT UNITA, IF AC11 (1) SELECT UNIT B
5739 6700 0000 XSEL, 0

```

```

5740 6201 6750 K67X0, 6750
5741 6202 5600 JMP I XSEL
5742 /THE CONTENTS OF THE AC AT ENTRY ARE THE CONTENTS OF PROGRAM LOCATION "TCOMMAND"
5743
5744 6203 6203 XLCD, .
5745 6204 3112 DCA COMMAND
5746 6205 1112 TAD COMMAND
5747 6206 6751 K67X1, 6751
5748 6207 5603 JMP I XLCD
5749 /LOAD THE COMMAND FOR: FILL BUFFER, AND EMPTY BUFFER
5750 /WITH THE RX01 INTERRUPT ENABLED
5751
5752 /FORM: (AC) IS COMMAND; LCDA; NORMAL RETURN; ERROR RETURN
5753
5754 6210 6210 XLCD, .
5755 6211 1777* TAD UNIT
5756 6212 3112 DCA COMMAND
5757 /THE CONTENTS OF THE AC WILL = 100 IF 8 BIT MODE
5758
5759 6213 7307 CLL CLA IAC RTL
5760 6214 0132 AND TESTP
5761 6215 7640 SZA CLA
5762 6216 1376 TAD (100)
5763 6217 1112 TAD COMMAND
5764 6220 3112 DCA COMMAND
5765 6221 1112 TAD COMMAND
5766 6222 4436 LCD
5767 6223 1210 TAD XLCD
5768 6224 3775* DCA RETURN
5769 6225 2210 ISZ XLCD
5770 6226 2210 ISZ XLCD
5771 6227 1374 TAD (PI)
5772 6230 3002 DCA IPI
5773 6231 7201 CLA IAC
5774 6232 4446 INTR
5775 6233 6001 ION
5776 6234 5610 JMP I XLCD
5777
5778 /LOAD THE COMMAND AND THE TRACK AND SECTOR ADDRESSES AND GO WITH INTERRUPT ENABLE I
5779
5780 /FORM: (AC) IS COMMAND; LCDB; NORMAL RETURN; ERROR
5781
5782 6235 6235 XLCD, .
5783 6236 1777* TAD UNIT / 0 OR 20
5784 6237 3112 DCA COMMAND /TEMPORARY STORAGE
5785
5786 /WHEN THE CONTENTS OF " GOBIT " ARE = 0 ; NO PROGRAM IRQ IS EXPECTED FROM THE DISK
5787
5788 6240 7307 CLL CLA IAC RTL / 4
5789 6241 0132 AND TESTP /TESTP FOR 8/12 BIT MODE SELECTION
5790 6242 7640 SZA CLA
5791 6243 1376 TAD (100) / 100 FOR 8 BIT MODE SELECTION
5792 6244 1112 TAD COMMAND / 4, 14 OR 6
5793 6245 3112 DCA COMMAND
5794 6246 1112 LCDB, TAD COMMAND

```

```

5795 6247 4436 LCD
5796 /
5797 /LOAD THE TRACK AND SECTOR ADDRESSES FOR THE COMMANDS;
5798 /
5799 /WRITE, OR WRITE DELETED DATA, OR READ SECTOR
5800 /
5801 6250 1373 TAD (LCDBRETURN)
5802 6251 3775 DCA RETURN
5803 6252 1372 TAD (NOP)
5804 6253 3315 DCA XLDBRETURN
5805 6254 7201 CLA IAC
5806 6255 4446 INTR /ENABLE RX01 INTERRUPT
5807 6256 4443 STR
5808 6257 5256 JMP ,-1 /SKIP ON TRANSFER REQUEST FLAG
5809 6260 1124 TAD TARGET
5810 6261 4442 XDROUT / SECTOR
5811 6262 4443 STR
5812 6263 5262 JMP ,-1 /SKIP ON TRANSFER REQUEST FLAG
5813 6264 1131 TAD TARGET
5814 6265 4442 XDROUT / TRACK
5815 6266 1374 TAD (PI)
5816 6267 3002 DCA IPI
5817 /
5818 /WAIT FOR A PROGRAM INTERRUPT REQUEST
5819 /
5820 / ; BUT WHILE WAITING, DISPLAY IN THE MQ THE CODED INFORMATION
5821 /AS INDICATED BY ACCUMULATOR SWITCHES 9, 10, AND 11 AT RUN-TIME
5822 /
5823 / 0 = TARGET TRACK AND SECTOR
5824 / 1 = TEST PARAMETERS (SELECTED FROM SA 200)
5825 / 2 = " A " STATUS
5826 / 3 = " COMMAND " WORD TO RX01
5827 /
5828 6270 6001 XPI, ION
5829 6271 1371 TAD (=40)
5830 6272 3116 DCA HANGER
5831 6273 4572 LAS
5832 6274 0370 AND (SW10+SW11)
5833 6275 1367 TAD (TAD I DISPLAY)
5834 6276 3277 DCA ,+1
5835 6277 1710 TAD I DISPLAY
5836 6300 7421 MQL
5837 6301 2153 ISZ H1
5838 6302 5273 JMP XPI+3
5839 6303 2116 ISZ HANGER
5840 6304 5273 JMP XPI+3
5841 6305 7200 CIA
5842 6306 1366 TAD (XPI)
5843 6307 5765 JMP HUNGUP
5844 6310 0151 DISPLAY, XTARGET
5845 6311 0132 TESTP
5846 6312 0126 ASTATUS
5847 6313 0112 COMMAND
5848 /RETURN TO HERE IF ANY ERROR OCCURS (OF IF "DONE" FLAG OCCURS)
5849 /

```

```

5850 /IF A PARITY ERROR OCCURS THEN RETRY TO LOAD THE COMMAND
5851 /
5852 /IF NOT A PARITY ERROR THEN THIS MUST BE THAT "DONE" FLAG I MENTIONED
5853 /
5854 6314 5326 LCDBRETURN, JMP OTHERERRORS / "JMP OTHERERRORS" IF RETURN IS OK
5855 6315 7000 XLDBRETURN, NOP
5856 6316 7305 CLL CLA IAC RAL
5857 6317 0126 AND ASTATUS
5858 6320 7650 SNA CLA
5859 6321 5326 JMP OTHERERRORS
5860 /PARITY ERROR = RETRY
5861 /
5862 6322 2146 ISZ PRETRY
5863 6323 5246 JMP LCDBL
5864 6324 4502 SCOPE
5865 6325 5453 EXIT
5866 /
5867 /THESE ARE ALL OTHER ERRORS WHICH MAY OCCUR
5868 /
5869 6326 1364 OTHERERRORS, TAD (JMP OTHERERRORS)
5870 6327 3315 DCA XLDBRETURN
5871 /
5872 6330 1373 TAD (LCDBRETURN)
5873 6331 7041 CIA
5874 6332 1775 TAD RETURN
5875 6333 1235 TAD XLDB
5876 6334 3235 DCA XLDB
5877 6335 5635 JMP I XLDB
5878 /TRANSFER DATA REGISTER (FROM) THE RX01 CONTROL
5879 /
5880 6364 5326
5881 6365 3341
5882 6366 6270
5883 6367 1710
5884 6370 0003
5885 6371 7740
5886 6372 7000
5887 6373 6314
5888 6374 5417
5889 6375 5517
5890 6376 0100
5891 6377 4242
5892 6400 6400 XXDRIN, .
5893 6401 6752 K67X2A, 6752
5894 6402 5600 JMP I XXDRIN
5895 /TRANSFER DATA REGISTER (TO) THE RX01 CONTROL
5896 /
5897 6403 6403 XXDROUT, .
5898 6404 6752 K67X2B, 6752
5899 6405 7200 CLA
5900 6406 5603 JMP I XXDROUT
5901 /SKIP ON TRANSFER REQUEST
5902 /
5903 6407 6407 XSTR, .

```

```

5904 6410 6753 K67X3A, 6753
5905 6411 5607 JMP I XSTR
5906 6412 2207 ISZ XSTR
5907 6413 5607 JMP I XSTR
5908
5909 6414 6414 XSER, .
5910 6415 6754 K67X4A, 6754
5911 6416 5614 JMP I XSER
5912 6417 2214 ISZ XSER
5913 6420 5614 JMP I XSER
5914 /SKIP ON RX01 DONE FLAG
5915 /
5916 6421 6421 XSDN, .
5917 6422 6755 K67X5A, 6755
5918 6423 5621 JMP I XSDN
5919 6424 2221 ISZ XSDN
5920 6425 5621 JMP I XSDN
5921 /ENABLE / DISABLE RX01 INTERRUPT ENABLE
5922 /
5923 / AC = 1 AT ENTRY TO ENABLE INTERRUPT
5924 /
5925 /FORM: (AC = 0, OR 1); INTR
5926 /
5927 / (GOBIT) = 0, NO RX01 PI IS EXPECTED
5928 /
5929 / (GOBIT) = 1, AN RX01 PI IS EXPECTED
5930 /
5931 6426 6426 XINTR, .
5932 6427 3154 DCA GOBIT
5933 6430 1154 TAD GOBIT
5934 6431 6756 K67X6, 6756
5935 6432 7200 CLA
5936 6433 5626 JMP I XINTR
5937 /INITIALIZE (POWER CLEAR) THE RX01 SUBSYSTEM
5938 /
5939 6434 6434 XINIT, .
5940 6435 6757 K67X7A, 6757
5941 /THE LABEL " SDNSECOND " MUST RESIDE HERE BECAUSE OF REFERENCES WITHIN " ERROR "
5942 /
5943 6436 4451 CKUNIT /GO CHECK IF ON UNIT A(RXA) OR UNIT B
5944 6437 4471 WAIT
5945 6440 4445 SDNSECOND, SDN
5946 6441 5237 JMP .-2
5947 6442 3006 DCA 6 /WAIT POINTER
5948 6443 4444 SER
5949 6444 5246 JMP XXINIT
5950 /AN ERROR HAS OCCURED FROM THE "INIT"
5951 /
5952 / [HOME] WAS THE TARGET
5953 /
5954 6445 4576 FORCE
5955 6446 3123 XXINIT, DCA SSTART
5956 6447 5634 JMP I XINIT
5957 /
5958 /

```

```

5959 /
5960 6450 6450 XINITB, .
5961 6451 6757 K67X7B, 6757
5962 6452 5650 JMP I XINITB
5963 REMOVE, TEXT "___REMOVE DIAGNOSTIC DISKETTE"
6454 2205
6455 1517
6456 2605
6457 4004
6460 1101
6461 0716
6462 1723
6463 2411
6464 0340
6465 0411
6466 2313
6467 0524
6470 2405
6471 0000
5964 6472 3737 MIDENTIFICATION,TEXT "___MAINDEC-08-DIRXA-D"
6473 1501
6474 1116
6475 0405
6476 0355
6477 6070
6500 5504
6501 1122
6502 3001
6503 5504
6504 0000
5965 MSELECT, TEXT "___SELECT PARAMETERS (INCLUDING DEVICE CODE)"
6505 3737
6506 2305
6507 1405
6510 0324
6511 4020
6512 0122
6513 0115
6514 0524
6515 0522
6516 2340
6517 5011
6520 1603
6521 1425
6522 0411
6523 1607
6524 4004
6525 0526
6526 1103
6527 0540
6530 0317
6531 0405
6532 5100
5966 MDIESTP, TEXT "___TEST PARAMETERS;"
6533 3737
6534 2405
6535 2324

```

	6436	4020						
	6437	0122						
	6440	0115						
	6441	0524						
	6442	0522						
	6443	2372						
	6444	4000						
5967	6445	3737	MXEHEADER,	TEXT	"_ERR FAT FAST	EAC	GOOD PASS"	
	6446	0522						
	6447	2240						
	6450	4006						
	6451	0174						
	6452	4040						
	6453	0601						
	6454	2324						
	6455	4040						
	6456	4040						
	6457	4040						
	6460	0501						
	6461	0340						
	6462	4007						
	6463	1717						
	6464	0440						
	6465	2001						
	6466	2323						
	6467	0000						
5968	6470	3727	MX2HEADER,	TEXT	"_WORD GOOD BAD"			
	6471	1722						
	6472	0440						
	6473	0717						
	6474	1704						
	6475	4002						
	6476	0104						
	6477	0000						
5969	6400	5440	MCOMMA,	TEXT	" , "			
	6401	0000						
5970	6402	3700	MCRLF,	TEXT	"_"			
5971	6403	3705	MEOT,	TEXT	"_END OF TEST "			
	6404	1604						
	6405	4017						
	6406	0640						
	6407	2405						
	6410	2324						
	6411	4000						
5972	6412	3704	MHUNGPC,	TEXT	"_DEVICE TEST HUNG AT PC "			
	6413	0576						
	6414	1103						
	6415	0540						
	6416	2405						
	6417	2324						
	6420	4010						
	6421	2516						
	6422	0740						
	6423	0174						
	6424	4020						

	6625	0340						
	6626	0000						
5973	6627	3737	MEHEADER,	TEXT	"_FAT CMND XDR CODE RSTA START	TARGET	TEST PASS"	
	6630	0601						
	6631	2440						
	6632	4003						
	6633	1516						
	6634	0440						
	6635	3004						
	6636	2240						
	6637	4003						
	6640	1704						
	6641	0540						
	6642	4022						
	6643	2324						
	6644	0140						
	6645	4023						
	6646	2401						
	6647	2224						
	6650	4040						
	6651	4040						
	6652	4024						
	6653	0122						
	6654	0705						
	6655	2440						
	6656	4040						
	6657	4024						
	6660	0523						
	6661	2440						
	6662	2001						
	6663	2323						
	6664	0000						
5974	6665	3310	MHOME,	TEXT	"[HOME]"			
	6666	1715						
	6667	0535						
	6670	0000						
5975	6671	1116	MINIT,	TEXT	"INIT"			
	6672	1124						
	6673	0000						
5976	6674	3737	MOD,	TEXT	"_OD = "			
	6675	1704						
	6676	4075						
	6677	4000						
5977	6700	4011	MID,	TEXT	" ID = "			
	6701	0440						
	6702	7540						
	6703	0000						
5978	6704	4006	MFIRST,	TEXT	" FIRST = "			
	6705	1122						
	6706	2324						
	6707	4075						
	6710	4000						
5979	6711	4014	MLAST,	TEXT	" LAST = "			
	6712	0123						
	6713	2440						

6714	7540		
6715	0000		
5980	6716	MWRITE,	TEXT "_WRITE="
	6717		
	6720		
	6721		
5981	6722	MREAD,	TEXT "_READ="
	6723		
	6724		
	6725		
5982	6726	MDNSEPROR,	TEXT "NO CRC BUT"
	6727		
	6730		
	6731		
	6732		
	6733		
5983	6734	MDWESERROR,	TEXT "CRC AND"
	6735		
	6736		
	6737		
5984	6740	MSNDERROR,	TEXT "CRC BUT NO DATA ERROR"
	6741		
	6742		
	6743		
	6744		
	6745		
	6746		
	6747		
	6750		
	6751		
	6752		
5985	6753	MDATAERROR,	TEXT " DATA ERROR_"
	6754		
	6755		
	6756		
	6757		
	6760		
	6761		
5986	6762	MWORD,	TEXT "WORD"
	6763		
	6764		
5987	6765	MBYTE,	TEXT "BYTE"
	6766		
	6767		
5988	6770	MGB,	TEXT " GOOD BAD"
	6771		
	6772		
	6773		
	6774		
5989	6775	MSUMCHECK,	TEXT "_SUM-CHECK IS "
	6776		
	6777		
	7000		
	7001		
	7002		

	7003	2340		
	7004	0000		
5990	7005	1713	MOK,	TEXT "OK"
	7006	0000		
5991	7007	3724	MDESUMMARY,	TEXT "_TOTAL BAD = "
	7010	1724		
	7011	0114		
	7012	4002		
	7013	0104		
	7014	4075		
	7015	4000		
5992	7016	7200	MCOLON,	TEXT ":"
5993	7017	5500	MDASH,	TEXT "-"
5994	7020	3715	MEDDDIDNOT,	TEXT "_MISSING DD MARK"
	7021	1123		
	7022	2311		
	7023	1607		
	7024	4004		
	7025	0440		
	7026	1501		
	7027	2213		
	7030	0000		
5995	7031	3725	MUDDID,	TEXT "_UNEXPECTED DD MARK"
	7032	1605		
	7033	3020		
	7034	0503		
	7035	2405		
	7036	0440		
	7037	0404		
	7040	4015		
	7041	0122		
	7042	1300		
5996	7043	3725	MSDNUNEXPECTED,	TEXT "_UNEXPECTED RX01 IRQ"
	7044	1605		
	7045	3020		
	7046	0503		
	7047	2405		
	7050	0440		
	7051	2230		
	7052	6061		
	7053	4011		
	7054	2221		
	7055	0000		
5997	7056	3715	MNOSE,	TEXT "_MISSING ERROR FLAG"
	7057	1123		
	7060	2311		
	7061	1607		
	7062	4005		
	7063	2222		
	7064	1722		
	7065	4006		
	7066	1401		
	7067	0700		
5998	7070	3737	MUNKNOWN,	TEXT "_UNKNOWN IRQ"
	7071	2516		


```

7072 1316
7073 1727
7074 1640
7075 1122
7076 2100
5999 7077 3704 MDEV, TEXT " _DEVICE CODE TO BE USED "
7100 0526
7101 1103
7102 0540
7103 0317
7104 0405
7105 4024
7106 1740
7107 0205
7110 4025
7111 2305
7112 0440
7113 4000

```

```

6000 /THE FOLLOWING IS THE WRITE BUFFER ALLOCATED STORAGE
6001 /
6002 7114 WBUFFER=
6003 7314 RBUFFER=WBUFFER+200
6004 7514 *RBUFFER+200
6005
6006
6007
6008
6009 0200 *200 /AUTO START BINARY
6010
6011 $$$

```

```

0000 11100000 11110000 11101111 11111111 11111111 11111111 11111111 11111111
0100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11111111 11111111 11111111 11000001 11111111 11111111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0500 11111111 11111111 11111111 11110000 00000000 00000000 00011111 11111111

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 11111111 11111111 11111111 11111111 11100000 00000000 01111111

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

1200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

1400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1500 11111111 11111111 11111111 11111111 11111111 11111111 11100000 01111111

1600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1700 11111111 11111111 11111111 11111111 11000000 00111111 11111111 11111111

2000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2100 11111111 11111111 11111111 11111111 11111111 11111110 00011111 11111111

2200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2300 11111111 11111111 11111111 11111111 11111111 00000011 11111111 11111111

2400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2500 11111111 11111111 11111111 11111111 11111111 11111110 00111111 11111111

2600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2700 11111111 11111111 11111111 11111111 11111111 11111100 00001111 11111111

3000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3100 11111111 11111111 11111111 11111111 11111111 11111110 00111111 11111111

3200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3300 11111111 11111111 11111111 11111111 11111111 11100000 00001111 11111111

3400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3500 11111111 11111111 11111111 11111111 11111111 11100000 00111111

3600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3700 11111111 11111111 11111111 11111111 11111111 11110011 11111111

```

```

4000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4100 11111111 11111111 11111111 11111111 11111111 11100000 00000000 11111111

4200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4300 11111111 11111111 11111111 11111111 11111111 11111111 11000000 00001111

4400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4500 11111111 11111111 11111111 11111111 11111111 11111111 11000000 00111111

4600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 10111111

5000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5100 11111111 11111111 11111111 11111111 11111111 11100000 01111111 11111111

5200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5300 11111111 11111111 11111111 11111111 11111111 11111111 10011111 11111111

5400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5500 11111111 11111111 11111111 11111111 11111110 00000000 00001111 11111111

5600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5700 11111111 11111111 11111111 11111111 11111111 10000000 00111111 11111111

6000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
6100 11111111 11111111 11111111 11111111 11111111 11100000 00001111 11111111

6200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
6300 11111111 11111111 11111111 11111100 00000000 00000000 00001111 11111111

6400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
6500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

6600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
6700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

7000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
7100 11111111 11110000 00000000 00000000 00000000 00000000 00000000 00000000

7200
7300

7400
7500

7600
7700
    
```

```

A10 0010 C8K212 0413 CKUNIT 4451 E2 0617
A11 0011 C8K215 0412 CLKCNT 4151 E20 0655
A12 0012 C8K240 0045 CNOTFI 3720 E21 0660
A13 0013 C8K260 0046 CNTVAL 0663 E210 4557
A14 0014 C8K275 0047 COMMAN 0112 E211 2320
ACL 7701 C8K277 0050 COMP 2746 E212 2332
ACTIVE 0741 C8K303 0414 COMPAR 3601 E22 0663
AERROR 4467 C8K307 0415 COMPRE 0113 E23 0666
ALT12 1400 C8K322 0051 COUNT 4152 E24 0671
ALT12L 1403 C8K323 0052 CRCERR 3456 E240 2030
ANDPET 4113 C8K336 0416 CSTATU 0130 E241 2070
APTR 4470 C8K7 0044 DASHAL 5725 E242 2101
APOUND 0233 C8K77 0407 DASHBC 5731 E245 2033
ASTATU 0126 C8LOPA 0231 DATAER 3643 E25 0674
BELL 0207 C8M10 0042 DCACII 0732 E26 0704
BITMOD 3763 C8M260 0043 DCADOA 2531 E27 0712
BLANK 0170 C8M3 0037 DCATAR 4332 E270 2135
BSTATU 0127 C8M4 0040 DDERRO 5537 E271 2140
BSW 4433 C8M40 0410 DDIGNO 5475 E272 2151
BSWAC 2623 C8M5 0041 DIGITS 5063 E28 0720
BSW1N 2624 C8MODE 0035 DISPLA 6310 E2PRE 0415
BSW1AL 2622 C8OCTA 4427 DMTYPE 6060 E3 0623
BUSY 0111 C8PASS 4426 DNS 3510 E30 0744
C87600 0417 C8PFLD 0543 DNSLOG 0143 E3PRE 0425
C8BLD 0400 C8PRMT 4426 DOA 2536 E40 1044
C8BY1 0640 C8PSW 0216 DOB 2537 E41 1060
C8CDF 0033 C8RET2 0526 DUCNT 0660 E42 1021
C8CDI 0032 C8RET3 4131 DONE 4454 E4PRE 0450
C8CHAR 0031 C8RET4 3317 DOPACK 0611 E50 1105
C8CK22 1060 C8RTN 0402 DOSET 0662 E51 1114
C8CKP 1057 C8SFLG 0404 DRVZRO 2743 E52 1123
C8CKSW 4425 C8SRLP 0252 DTESTP 0114 E53 1111
C8CNTL 0420 C8STAR 0200 DTYPE 5535 E54 1120
C8CNTR 0036 C8STRT 0401 DWESER 3700 E55 1127
C8CRLF 0523 C8SWR 0034 DWS 3510 E56 1102
C8CTLG 0465 C8TEMP 0030 DWSLOG 0142 E60 1142
C8CTLQ 0457 C8TEST 0337 E0 0607 E61 1154
C8CTLQ 0455 C8TMP1 1056 EOPRE 0412 E62 1157
C8CTL8 0437 C8TTY 0514 E1 0613 E63 1162
C8D01 1005 C8TYP 0502 E10 0647 E70 1212
C8D04 1043 C8WAIT 0444 E100 1226 E7000 2405
C8ECHO 0475 CAF 6007 E11 0632 E7001 2416
C8ENTR 0200 CCNT 5066 E110 1252 E7002 2432
C8EPR 0316 CHARLI 5320 E120 1311 E7003 2446
C8EXT1 0531 CHECKC 4424 E121 1315 E7004 2457
C8EXT2 0541 CHECKU 6151 E122 1321 E7005 2511
C8EXTB 0707 CHEK22 4472 E123 1326 E7006 2473
C8FLG 0403 CHGMSK 2750 E124 1276 EA120 1412
C8GET1 0704 CHNDEV 5067 E130 1513 EA121 1416
C8K100 0411 CKCOU 0642 E131 1526 EA122 1440
C8K177 0405 CKSFLG 5400 E140 1661 EA123 1451
C8K200 0406 CKSWIT 4572 E1PRE 0445 EAC 0167
    
```

EB	1622	INMODE	0351	MDASH	7017	NUREAD	3521
EBCOMP	1630	INSUMC	3762	MDATAE	6753	NUWORD	5253
EBCOOP	1635	INTR	4446	MDESUM	7007	NXTUNT	0523
EBOW	1663	IOF	6002	MDEV	7077	OD	0027
ECCOMA	0152	ION	6001	MDNSER	6726	OK	4465
EERROR	5646	IP1	0002	MDTEST	6533	OKSTAR	4643
EMPT	3617	IRDWR	2000	MDWESE	6734	ONECRL	5671
EMPTYE	4011	ISZCOM	3757	MEDDD1	7020	OTERR	6326
EMPTYL	4000	ISZDIG	5042	MEHEAD	6627	OUTPUT	4263
EMPTYO	4022	JMPDIG	5032	MEDT	6603	PACKDO	4531
END-OM	3636	JMPICO	3600	MESPAS	0664	PASCNT	0661
EPCSCO	1364	JMPWHI	3263	MESSAG	5262	PASS	0163
EPEUR	5520	K0007	0174	MFIRST	6704	PAT2	3125
EPRNK	4452	K6500	4150	MGB	6770	PAT3	3126
ERRNPS	1365	K6520	4361	MHOME	6665	PAT4	3127
EXIT	5453	K67X0	6201	MHUNG	6612	PAT5	3130
EXITCK	4534	K67X1	6206	MID	6700	PAT6	3140
FB	1472	K67X2A	6401	MIDENT	6472	PATSUM	0155
FB128B	4541	K67X2B	6404	MINIT	6671	PATTER	3115
FBEA	1477	K67X2C	3623	MLAST	6711	PCLF	6662
FLU	3221	K67X3A	6410	MNOSER	7056	PCSCUP	1366
FILLER	3226	K67X3B	0707	MOD	6674	PHOME	6032
FILUOK	3237	K67X4A	6415	MOK	7005	PI	5417
FIRST	0031	K67X4B	0715	MURETE	0452	PIEXIT	5510
FIRSTE	0115	K67X5A	6422	MQA	7501	PISON	5434
FIRSTT	0453	K67X5B	0701	MQL	7421	PNTXT	5341
FORPE	4576	K67X6	6431	MREAD	6722	PNTID	1531
GEMTES	3131	K67X6B	3622	MSB	0161	PULL	4237
GTAAPA	4455	K67X6C	3637	MSDNUN	7043	PRETES	0400
GTAAP	4456	K67X7A	6435	MSELEC	6505	PRETRY	0146
GTAATR	4457	K67X7B	6451	MSNDER	6740	PRINT	4473
GTHINI	4460	K7000	0173	MSUMCH	6775	PSIE	6665
GUBTT	0154	K7377	1561	MUDDDI	7031	PSKE	6663
GNOP	0166	K7777	0117	MUNKNO	7070	PSKF	6661
GTF	6004	KPETRY	0121	MWORD	6762	PSIB	6664
HT	0153	LAS	4572	MWRITE	6716	PTSTUR	1033
HALT	4461	LASSW0	6111	MK	0522	Q46ID	4505
HANGER	0116	LASSW4	5642	MX2HEA	6570	Q46UD	4513
HLT	4461	LAST	0032	MXEHEA	6545	Q40D	4454
HLT16	6113	LCD	4436	NEXT	4210	Q60D	4500
HLTA	4101	LCDA	4437	NEXTAC	0746	QUIET	3335
HLT7	3507	LCDB	4440	NEXTSE	4625	R1	4726
HLTNGP	1562	LCDBL	6246	NOHEAD	5655	R1RETR	0140
HUNGPC	3353	LCDBP	6314	NOMORE	0465	R2	4727
HUNGUP	3341	LOCKUP	4464	NOPRIN	6061	R2RETR	0141
ID	0030	LSB	0160	NORX01	0431	RANGEN	4712
IF	4710	MASK	0136	NOSEOP	1340	RBUFFE	7314
INIT	4447	MASK1	2747	WOSER	3322	RDC	0120
INITB	4450	MBITE	6765	WOSER	0652	RDURWR	3552
INITSE	4462	MCOLON	7016	WOTCL8	4537	READ	4474
INITSW	6115	MCUMMA	6600	WOTEST	5453	READCU	4475
INITTR	4463	MCRLF	6602	NULINE	5302	READER	3441

RFAOL	3422	SW0	4000	TADLAS	3273	XAC	5515
REANOK	3531	SW1	2000	TARGET	0131	XAERRO	4346
REANRE	3425	SW10	0002	TEST	0171	XANDRE	4120
REBPGI	0511	SW11	0001	TEST1	2524	XAPT8	1545
RFCALI	6073	SW2	1000	TEST2	2522	XASTAT	3761
RFFILL	3214	SW3	0400	TEST3	2516	XBSW	2600
RFMVNE	6453	SW4	0200	TEST4	2514	XC	0300
RFRPAD	3432	SW5	0100	TEST5	2526	XC8CHA	0350
RFSPEK	3434	SW6	0040	TESTP	0132	XC8ENT	4425
RESFOU	4274	SW7	0020	TESTS	4730	XC8OCT	1035
RETHRN	5517	SW8	0010	TESTX	2532	XC8PAS	0600
REWRIT	3206	SW9	0004	TESTXL	2535	XC8PNT	1000
RURGGE	3547	T0	0600	THEL	3003	XC8SW	0673
ROK	3543	T1	0626	THEXEX	4114	XCHECK	4516
RST	4476	T10	1215	THETES	3000	XCHKZ2	5144
RSTART	0200	T11	1233	TICK	4466	XCKSWI	3400
RSTR	4477	T12	1260	TRACKS	0147	XCKUNT	2335
RXERRO	5600	T13	1456	TSTUNT	2732	XCNT	0175
RXHEBS	0165	T14	1603	TTRACK	0150	XCOMPA	3663
SAVEBS	5613	T15	1601	TYBUS	0162	XCR CER	3505
SAVECS	5626	T16	1455	TWOCRL	5667	XD	0400
SCOE	4502	T17	1602	TX	2303	XDEVIC	5133
SCODIN	1357	T2	0652	TY1ASC	5321	XDDEE	4243
SDN	4445	T20	1600	TY4OCT	4505	XDRIN	4441
SDNEEC	6440	T20STR	2323	TY8OCT	4434	XDRGUT	4442
SDNHNE	3320	T20XDR	2310	TYASTA	5740	XEMPTY	4072
SECTOR	0122	T21	1675	TYCSTA	6000	XERROR	2625
SEFL	4435	T22	1674	TYPEIT	4506	XFLENG	3715
SELHNT	6127	T23	2265	UNIT	4242	XFORCE	5530
SEQ	4276	T24	1714	UNITCK	0135	XGETAP	3035
SEQ000	4306	T25	1716	UNITIS	4235	XGETAS	4607
SEQ001	4317	T26	1720	UNITX	4241	XGETAT	4261
SEQ010	4400	T27	2107	UNITZ	4240	XGETUN	4200
SEQ100	4444	T3	0723	UNKNOW	5505	XHALT	4121
SEQ111	4330	T30	2200	UREAD	3524	XHUNG	3344
SEQ1	4422	T31	2202	VERIFY	5500	XI	1100
SEQ6	4465	T32	2223	VT78CK	1165	XINIT	6434
SER	4444	T33	2241	WAIT	4471	XINITB	6450
SETHNT	6146	T34	2300	WAITTY	4507	XINITIS	4600
SETHP	4500	T35	2276	WBUFFR	7114	XINITT	4250
SHIPT	5015	T36	2277	WHICHR	3267	XINTR	6426
SHIPT5	5065	T37	2275	WNODOK	3264	XK67X2	5135
SND	3510	T4	1000	WORDX	0156	XK67X3	5136
SNDLOG	0144	T4B	1004	WORDY	0157	XK67X4	5137
SPECIA	4501	T5	1071	WRESEE	3212	XK67X5	5140
SRETRY	0145	T6	1132	WRITE	4510	XK67X6	5141
SSTART	0123	T7	1200	WRITEL	3204	XK67X7	5142
STARGE	0124	T7OK	1213	WRITED	3257	XKCC	0323
START	0125	TAB	4504	WRITER	3246	XLCD	6203
STODNT	1034	TAB12	5712	XUNITS	4236	XLCDA	6210
STR	4443	TAB31	6002	XAI0	0133	XLCDB	6235
SUBRCO	4503	TAB43	6022	XAI1	0134	XLCDBR	6315

XLINK	5516	XXX	0137
XLOCKU	2545	XYTHEL	3026
XMESSA	3334	ZERO	5057
XX	0657		
XNOPRI	2715		
XNULIN	5311		
XDOCTAL	5062		
XOK	6123		
XOUTPU	5263		
XPAT6	3145		
XPATTE	3103		
XPI	6270		
XPRINT	5244		
XPDC	3420		
XREAD	3416		
XREADC	3412		
XRETUR	5521		
XRST	2441		
XRSTB	2400		
XSCOPE	1333		
XSDM	6421		
XSEL	6200		
XSEQ	4333		
XSEQ2	4413		
XSEQ3	4433		
XSER	6414		
XSETUP	4103		
XSHFTT	5020		
XSPFCI	5352		
XSSOP	1344		
XSTARG	4711		
XSTR	6407		
XTAB	5215		
XTABL	5221		
XTABLE	0151		
XTCF	5344		
XTHFL	3016		
XTIK	4133		
XTIAC	5000		
XTYAUC	5200		
XTYPEI	5305		
XWATT	1722		
XWATTT	5412		
XWHITE	3200		
XDXMP	5064		
XDXPIN	6400		
XDXDOU	6403		
XXFRC	5525		
XYGFTA	4610		
XXIMLT	6446		
XXTAB	5243		
XXTHEL	3024		

ERRORS DETECTED: 0
LINKS GENERATED: 179
RUN-TIME: 10 SECONDS
3K CORE USED

XDROUT	063#	2410	2728	2824	3676	4738	5810	5814	
XEMPTY	4773	4279	4303	4311	4326	4333	4337#		
XERROR	082	3336#		3341	3420				
XFLENG	4181	4187	4189#						
XFORCE	5381#	5642							
XGETAP	088	3502#	3516	3521	3567	3623			
XGETAS	090	4770#	4870						
XGETAT	092	3473	4484#	4566					
XGETUN	094	4411#	4440						
XHALT	097	4367#	4377	4380					
XHUNG	3793#	5425	5524						
XI	2718#	2223							
XINIT	074	5939#	5956						
XINITR	076	5960#	5962						
XINITB	099	4758#	4764						
XINITT	1001	4463#	4471						
XINTR	072	5931#	5936						
XK67X2	5018	5020	5039#						
XK67X3	5022	5040#							
XK67X4	5024	5041#							
XK67X5	5026	5042#							
XK67X6	5028	5030	5043#						
XK67X7	5032	5044#							
XKCC	1270#	5257							
XLCD	056	5744#	5748						
XLDA	058	5754#	5767	5769	5770	5776			
XLDB	060	5782#	5875	5876	5877				
XLDBH	5804	5855#	5870						
XLINK	5254	5351	5357#						
XLOCKU	1003	3284#	3295						
XMESSA	3770	3776	3779#						
XX	402	435#							
XNOPRJ	3351	3407#							
XNULIN	5163	5168#							
XOCTAL	4943	4944	4945	4951	4958	4959	4986#		
XOK	1006	5698#	5700	5701					
XOUTPU	5146#	5160	5161						
XPAT6	3614#	3622							
XPATL	3650	3551#	3566						
XPI	5828#	5838	5840	5842					
XPRINT	1018	4349	5130#	5134	5135	5195	5197		
XRDC	3833	3851#							
XREAD	1020	3017	3832	3849#	4006	4015			
XREADC	1022	3830#	3831						
XRETRK	5343	5369#							
XRST	1024	3186#	3242						
XRSTL	1026	3144#	3176						
XSCOPE	1032	2227#	2239	2241	2246	2249	2257	2261	
XSDN	070	5916#	5918	5919	5920				
XSET	054	5739#	5741						
XSEQ	4652#	4631	4698						
XSEQ2	4694	4604#							
XSEQ3	4617	4623#							

SEQ 0166

XSEF	068	5909#	5911	5012	5913				
XSETUP	1028	4346#	4350	4352	4353				
XSHIPT	4052#	4957							
XSPECI	1030	5209#	5211	5212	5213				
XSSCOP	1034	2244#	2245						
XSTARG	4763	4797	4801	4802	4809	4810	4812	4873#	
XSTP	066	5903#	5905	5906	5907				
XTAB	1036	5098#	5100	5104	5114				
XTABL	5102#	5111							
XTAPGE	1110#	4845	5844						
XTCF	1094	5200#	5266						
XTHEL	3472	3483#							
XTICK	1008	2642	4384#	4387	4389	4396			
XY40C	1038	4936#	4940	4941	4942	4973			
XY80C	052	5081#	5082	5084	5092	5093			
XYYPEI	1040	5162	5164#	5167	5174	5204			
XWAIT	1014	2678#	2685	2687	2688	2692	5690		
XWAITT	1042	5245#	5249						
XWRITE	1044	2924	3642#	3682	3688	3724			
XXDMP	4963	4966	4988#						
XXDRIH	062	5892#	5894						
XXDPOU	064	5897#	5900						
XXFORC	1150	5284	5378#	5385	5424	5428	5485	5493	5509
XXGETA	4774#								
XXINIT	5494	5510	5602	5949	5955#				
XXTAB	5101	5103	5105	5109	5113	5117	5120#		
XXTHEL	3491#	3498							
XXX	1096#	3552	3561	3563	3617	3618	3619	4016	4017
XYTHEL	3493#	3496						4573	4578
ZERO	4976	4983#						5396	5401
.L0357	1094	1301#							
.L0360	1092	1302#							
.L0361	171	201#	1272	1303#					
.L0362	168	202#	1271	1304#					
.L0363	165	203#	1260	1305#					
.L0364	162	204#	1253	1306#					
.L0365	160	205#	1234	1307#					
.L0366	157	206#	1221	1308#					
.L0367	154	182	207#	1219	1309#				
.L0370	153	186	190	208#	1203	1310#			
.L0371	123	125	127	138	156	173	196	209#	1202
.L0372	121	197	210#	1201	1312#				1311#
.L0373	116	211#	1195	1197	1313#				
.L0374	113	212#	1192	1314#					
.L0375	109	213#	1184	1315#					
.L0376	107	214#	1175	1316#					
.L0377	104	215#	1173	1317#					
.L0563	1490	1510#							
.L0564	1485	1511#							
.L0565	1484	1512#							
.L0566	1464	1513#							
.L0567	1456	1514#							
.L0570	1454	1480	1515#						

SEQ 0167

.L0571	344	356	361#	1452	1516#				
.L0572	781	362#	1438	1517#					
.L0573	768	363#	1401	1518#					
.L0574	247	364#	1382	1519#					
.L0575	244	365#	1379	1520#					
.L0576	241	366#	1344	1437	1521#				
.L0577	238	265	367#	1342	1433	1522#			
.L0767	462	467#							
.L0770	455	468#							
.L0771	416	469#	1758	1763#					
.L0772	413	470#	1745	1764#					
.L0773	400	471#	1743	1765#					
.L0774	397	405	472#	1741	1766#				
.L0775	395	407	473#	1622	1767#				
.L0776	387	409	474#	1609	1617	1625	1653	1768#	
.L0777	384	475#	1553	1769#					
.L1165	574	584#							
.L1166	572	585#							
.L1167	551	586#							
.L1170	550	587#							
.L1171	547	588#							
.L1172	521	589#							
.L1173	520	590#	1999	2004#					
.L1174	518	591#	1945	2005#					
.L1175	515	592#	1839	2006#					
.L1176	513	522	552	593#	1832	1868	2007#		
.L1177	504	594#	1802	1902	1947	2008#			
.L1367	2224	2272#							
.L1370	2223	2273#							
.L1371	2155	2274#							
.L1372	2146	2275#							
.L1373	2122	2276#							
.L1374	2097	2277#							
.L1375	2074	2278#							
.L1376	2035	2279#							
.L1377	2033	2077	2116	2168	2280#				
.L1571	2485	2488#							
.L1572	2405	2489#							
.L1573	2403	2490#							
.L1574	2400	2491#							
.L1575	2391	2396	2431	2433	2492#				
.L1576	2318	2493#							
.L1577	2316	2494#							
.L1752	2693	2695#							
.L1753	2691	2696#							
.L1754	2689	2697#							
.L1755	2652	2698#							
.L1756	2649	2651	2653	2699#					
.L1757	2644	2700#							
.L1760	2642	2701#							
.L1761	2640	2702#							
.L1762	2632	2703#							
.L1763	2629	2704#							

SEQ 0168

.L1764	2616	2705#							
.L1765	2615	2706#							
.L1766	2589	2707#							
.L1767	2578	2708#							
.L1770	2563	2609	2709#						
.L1771	2548	2710#							
.L1772	2538	2711#							
.L1773	2534	2645	2712#						
.L1774	2532	2550	2713#						
.L1775	2529	2714#							
.L1776	2525	2557	2584	2586	2715#				
.L1777	2519	2716#							
.L2163	2852	2865#							
.L2164	2849	2866#							
.L2165	2816	2867#							
.L2166	2808	2862	2868#						
.L2167	2797	2869#							
.L2170	2794	2870#							
.L2171	2776	2871#							
.L2172	2770	2806	2860	2872#					
.L2173	2765	2873#							
.L2174	2763	2874#							
.L2175	2759	2875#							
.L2176	2721	2876#							
.L2177	2718	2877#							
.L2356	3082	3122#							
.L2357	3075	3123#							
.L2360	3073	3124#							
.L2361	3065	3069	3125#						
.L2362	3036	3126#							
.L2363	3033	3127#							
.L2364	3018	3128#							
.L2365	3017	3129#							
.L2366	3016	3130#							
.L2367	2970	2989	3131#						
.L2370	2954	3132#							
.L2371	2931	2975	3028	3133#					
.L2372	2925	3134#							
.L2373	2924	3135#							
.L2374	2923	3136#							
.L2375	2903	3137#							
.L2376	2900	2988	3138#						
.L2377	2885	3139#							
.L2562	3286	3296#							
.L2563	3268	3297#							
.L2564	3258	3262	3298#						
.L2565	3256	3264	3299#						
.L2566	3232	3300#							
.L2567	3229	3301#							
.L2570	3218	3302#							
.L2571	3215	3303#							
.L2572	3187	3304#							
.L2573	3173	3305#							

SEQ 0169

.L2574	3164	3292	3306#			
.L2575	3163	3167	3214	3260	3266	3307#
.L2576	3159	3201	3308#			
.L2577	3145	3309#				
.L2764	3449	3453#				
.L2765	3447	3454#				
.L2766	3438	3455#				
.L2767	3435	3456#				
.L2770	3429	3457#				
.L2771	3428	3458#				
.L2772	3426	3459#				
.L2773	3408	3460#				
.L2774	3349	3461#				
.L2775	3345	3462#				
.L2776	3343	3463#				
.L2777	3319	3464#				
.L3162	3616	3624#				
.L3163	3551	3570	3625#			
.L3164	3547	3626#				
.L3165	3543	3627#				
.L3166	3538	3628#				
.L3167	3532	3629#				
.L3170	3531	3630#				
.L3171	3525	3631#				
.L3172	3517	3632#				
.L3173	3514	3633#				
.L3174	3513	3634#				
.L3175	3479	3482	3635#			
.L3176	3473	3636#				
.L3177	3472	3637#				
.L3364	3797	3802#				
.L3365	3783	3803#				
.L3366	3772	3804#				
.L3367	3769	3805#				
.L3370	3767	3806#				
.L3371	3762	3807#				
.L3372	3739	3808#				
.L3373	3732	3809#				
.L3374	3731	3810#				
.L3375	3684	3707	3811#			
.L3376	3666	3812#				
.L3377	3649	3813#				
.L3572	4018	4022#				
.L3573	4072	4023#				
.L3574	4030	4024#				
.L3575	4022	3997	4025#			
.L3576	4019	3994	4026#			
.L3577	3887	3954	4027#			
.L3766	4220	4244#				
.L3767	4205	4245#				
.L3770	4196	4246#				
.L3771	4186	4247#				
.L3772	4180	4248#				

SEQ 0170

.L3773	4152	4249#				
.L3774	4133	4250#				
.L3775	4064	4251#				
.L3776	4062	4252#				
.L3777	4060	4253#				
.L4170	4349	4401#				
.L4171	4348	4402#				
.L4172	4341	4345	4403#			
.L4173	4301	4404#				
.L4174	4298	4405#				
.L4175	4296	4299	4309	4315	4339	4406#
.L4176	4275	4407#				
.L4177	4256	4408#				
.L4374	4542	4585#				
.L4375	4541	4586#				
.L4376	4491	4587#				
.L4377	4487	4588#				
.L4572	4734	4749#				
.L4573	4707	4750#				
.L4574	4631	4698	4751#			
.L4575	4610	4752#				
.L4576	4609	4753#				
.L4577	4597	4754#				
.L4772	4859	4864	4927#			
.L4773	4853	4928#				
.L4774	4852	4863	4929#			
.L4775	4808	4930#				
.L4776	4803	4931#				
.L4777	4785	4932#				
.L5161	5056	5062#				
.L5162	5033	5063#				
.L5163	5031	5064#				
.L5164	5029	5065#				
.L5165	5027	5066#				
.L5166	5025	5067#				
.L5167	5023	5068#				
.L5170	5021	5069#				
.L5171	5019	5070#				
.L5172	5008	5071#				
.L5173	5002	5072#				
.L5174	4997	5073#				
.L5175	4982	5074#				
.L5176	4980	5075#				
.L5177	4967	4983	5076#			
.L5363	5202	5216#				
.L5364	5192	5217#				
.L5365	5184	5218#				
.L5366	5172	5219#				
.L5367	5170	5220#				
.L5370	5168	5221#				
.L5371	5155	5222#				
.L5372	5152	5223#				
.L5373	5151	5190	5224#			

SEQ 0171

