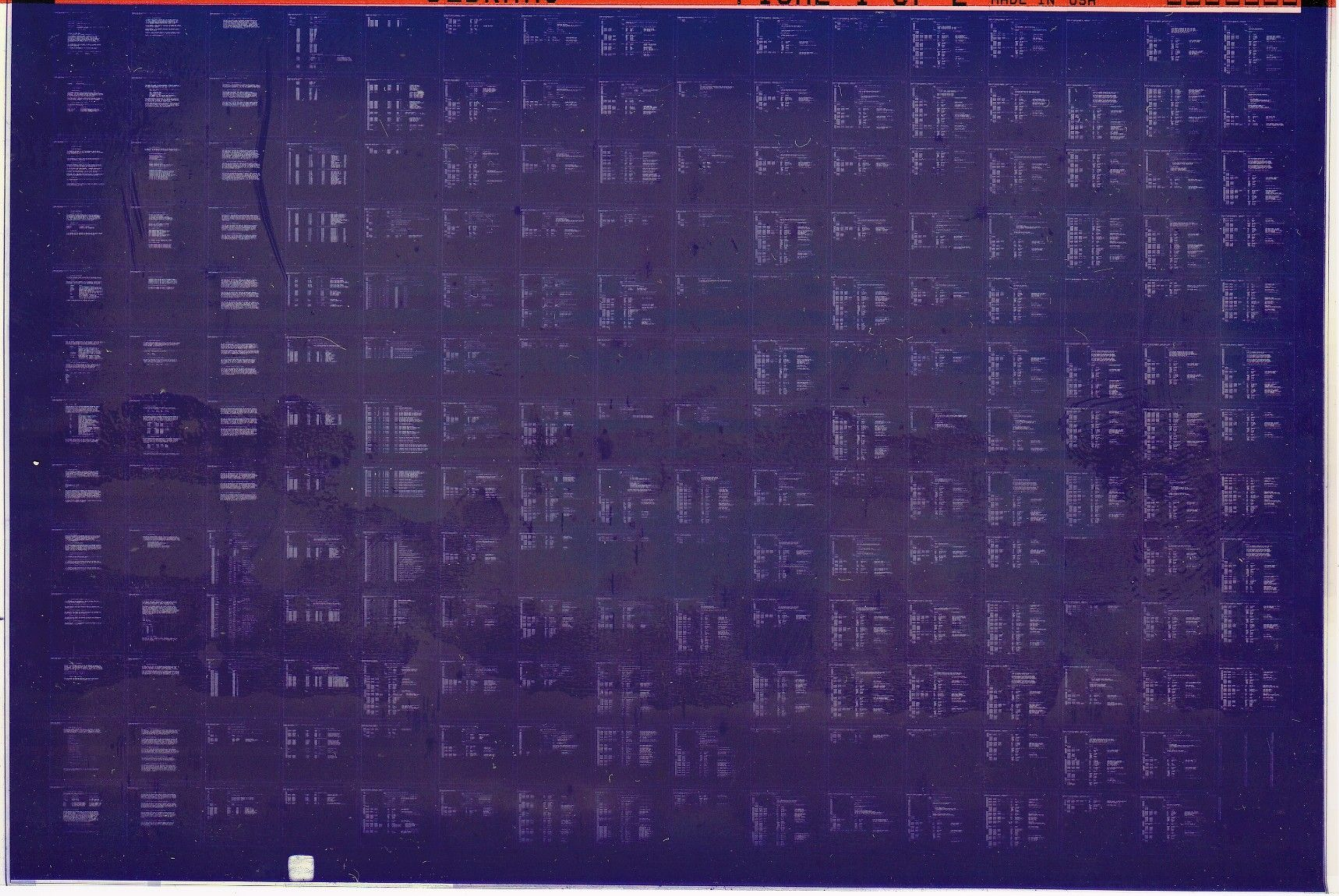
DR70-A,B,C

DR70 REPAIR DIAG
CZDRMA0

AH-T420A-MC
FICHE 1 OF 2



DR70-A,B,C

DR70 REPAIR DIAG
CZDRMAO

AH-T420A-MC
FICHE 2 OF 2

.REM &

IDENTIFICATION

PRODUCT CODE: AC-T419A-MC

PRODUCT NAME: CZDRMA0 DR70 REPAIR DIAG

PRODUCT DATE: FEBRUARY 25, 1983

MAINTAINER: CSS/GOVERNMENT NETWORKS GROUP DIAGNOSTIC ENGINEERING
COSTA MESA, CALIFORNIA

AUTHOR: H. PAUL HOLSINGER

0.0 MODIFICATION HISTORY

25-JAN-83 INITIAL DESIGN
23-FEB-83 REVISION A.0 SUBMITTED FOR RELEASE

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS PROGRAM CONTAINS A SET OF TESTS WHICH WILL VERIFY THE INTEGRITY OF THE DR70 INTERFACE CABLED TO AN RH11 OR RH70 MASSBUS CONTROLLER. THE TESTS REQUIRE THAT THE USER INTERFACE TO THE DR70 BE DISCONNECTED AND THE M8432 TEST MODULE AND CABLING BE INSTALLED. TESTING THE INTERFACE IS LIMITED TO ONE UNIT ONLY.

THE DEFAULT SET OF TESTS WILL EXERCISE BOTH REVISION "A" AND "B" INTERFACE BOARDS. ADVANCED TEST(S) MAY BE SELECTED DEPENDING UPON THE OPERATING REQUIREMENTS OF THE DR70 INTERFACE ENVIRNMENT.

1.2 RELATED DOCUMENTS AND STANDARDS

THE USER SHOULD REFER TO THE FOLLOWING MANUALS/OPTION DESCRIPTIONS FOR MORE DETAILED INFORMATION.

DOCUMENT	REV	DESCRIPTION
CSS-WO-F-5.2-26		RH70 MASSBUS CONTROLLER OPTION DES.
CSS-WO-F-5.2.27		RH11 MASSBUS CONTROLLER OPTION DES.
YW-C084C-00	P1	DR70B MASSBUS CHANNEL INTERFACE OPTION DES.
AC-S296A-AC	A0	CIQPMA0 XXDP+ PROGRAMMER'S MANUAL

2.0 HARDWARE REQUIREMENTS

2.1 SYSTEM REQUIREMENTS

THE PROGRAM IS DESIGNED TO TEST THE OPTION ONLY ON THE PDP-11 FAMILY PROCESSORS, WITH AT LEAST 16K MEMORY. PRIOR TO INSTALLATION OF THE DR70 INTERFACE, THE SYSTEM MUST HAVE EITHER AN RH11 OF RH70 MASSBUS CONTROLLER INSTALLED.

2.2 MEDIA

THIS DIAGNOSTIC IS AVAILABLE FROM THE SOFTWARE DISTRIBUTION CENTER (SDC) ON MULTIMEDIA. REFER TO THE HARDWARE DOCUMENTATION KIT HANDBOOK FOR ORDERING INFORMATION.

2.3 DIAGNOSTIC HIERARCY PREREQUISITES

BEFORE THE PROGRAM IS LOADED, THE RH11(RH70) MASSBUS CONTROLLER, THE DR70 AND THE M8432 TEST MODULE AND CABLING SHOULD BE INSTALLED ON THE PDP SYSTEM, AND CHECKED USING A FEW QUICK TESTS.

THE SYSTEM SHOULD BOOT XXDP+ PROPERLY: IMPROPER INSTALLATION MAY CAUSE THE BUS TO HANG, AND PREVENT ANY I/O FROM TAKING PLACE.

RH11(RH70) REGISTER ADDRESSING: THE RH11(RH70) REGISTERS SHOULD BE EXAMINED WITHOUT ADDRESS ERROR USING THE SWITCH REGISTER.

DR70 DRIVE TYPE: THE DRIVE TYPE REGISTER (CSR + 26) CONTENTS SHOULD BE EXAMINED TO DETERMINE THAT THE DR70 INTERFACE IS PRESENT ON THE MASSBUS, AND THE PROPER UNIT SELECT SWITCHES HAVE BEEN SET.

2.4 ASSUMPTIONS

ALL MASSBUS AND UNIBUS CABLING AND TERMINATORS ARE ASSUMED TO BE FUNCTIONING PROPERLY.

2.5 RESTRICTONS

THIS DIAGNOSTIC SUPPORTS TESTING OF A SINGLE DR70 CABLED TO A DEDICATED MASSBUS CONTROLLER. THE USER INTERFACE MUST BE REMOVED AND REPLACED BY THE M8432 MAINTENANCE CARD.

3.0 SOFTWARE REQUIREMENTS

3.1 DIAGNOSTIC SUPERVISOR

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 4 OF THIS DOCUMENT.

3.2 REQUIRED FILES

THE DISTRIBUTION KIT SHOULD CONTAIN THE FOLLOWING FILES:

ZDRMA0.BIN	DIAGNOSTIC PROGRAM
ZDRMA0.HLP	DIAGNOSTIC HELP INFORMATION
DRS.HLP	SUPERVISOR HELP INFORMATION

3.3 ASSUMPTIONS

IT IS ASSUMED THAT THE OPERATOR IS FAMILIAR WITH THE PDP-11 SYSTEM USING THE DR70 OPTION, AND THAT THE OPERATOR HAS READ THIS PROGRAM DOCUMENTATION BEFORE ATTEMPTING TO LOAD THE DIAGNOSTIC.

4.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES.
FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

4.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES
(SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY
BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 7.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 4.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 4.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO
YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

4.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 4.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

```
START/TESTS:1-5/PASS:1000/EOP:100
```

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

4.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

* ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

4.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

```
CHANGE HW (L)  ?

# UNITS (D)  ?

UNIT 0
DEVICE ADDRESS (O)  172440 ?
VECTOR ADDRESS (O)  224 ?
PRIORITY       (D)  5 ?
DRIVE          (D)  0 ?
```

UPON INITIAL LOADING OF THE PROGRAM, THE OPERATOR MAY ELECT TO UTILIZE THE ABOVE DEFAULT P-TABLE VALUES, WHICH HAVE ALREADY BEEN ASSEMBLED INTO THE PROGRAM. ADDITIONALLY, THE OPERATOR MAY CHOOSE TO MODIFY THE DEFAULT P-TABLE VALUES BY ANSWERING THE HARDWARE QUESTIONS TO BUILD THE HARDWARE P-TABLE FOR UNIT 0. THE OPERATOR MUST SPECIFY BOTH THE DEVICE AND INTERRUPT VECTOR ADDRESSES OF THE INTERFACE, WHICH ARE SELECTED BY THE APPROPRIATE JUMPERS ON THE RH11 OR RH70 CONTROLLER ADDRESS BOARD. IN ADDITION, THE OPERATOR MUST SPECIFY THE BUS REQUEST INTERRUPT PRIORITY AND THE DRIVE NO. SELECTED FOR THE INTERFACE.

THE DEFAULT VALUES PRESENT ARE PROVIDED TO AVOID CONFLICT WITH THE STANDARD PDP-11 SYSTEMDEVICE ADDRESSES. CONSULT THE PDP-11 PERIPHERALS HANDBOOK FOR THE STANDARD ASSIGNMENTS.

4.5 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y". THE SOFTWARE QUESTIONS AND THE DEFAULT VALUES ARE DESCRIBED IN THE NEXT PARAGRAPH(S).

4.5.1 SOFTWARE PARAMETER CONTROL

THE SOFTWARE QUESTION PARAMETERS ARE USED BY THE PROGRAM TO BUILD A TEST SELECTION WORD, WHICH CONTROLS DEFAULT AND OPTIONAL INTERFACE TESTING. THIS APPROACH ALLOWS EASY MODIFICATION OF INDIVIDUAL TEST EXECUTION CRITERIA, IF REQUIRED. THE TEST SELECTION MASK IS REBUILT EACH PASS, AND MAY BE MODIFIED AFTER ANY START, RESTART OR CONTINUE COMMAND. NOTE THAT ALL PARAMETER DEFAULT VALUES ARE INITIALLY CLEARED, WHICH OMITS ALL OPTIONAL TESTING.

4.5.2 SORTWARE QUESTION DESCRIPTIONS

(1) REVISION "A" INTERFACE (L) N ?

A "Y" RESPONSE TO THIS QUESTION WILL ADD/OMIT ALL TEST SECTIONS WHICH APPLY TO THE REVISION "A" DR70 INTERFACE.

(2) AT0 ATTN DISABLED [SW4 "OFF"] (L) N ?

A "Y" RESPONSE TO THIS QUESTION WILL BOTH ADD THE OPTIONAL TEST FOR SWITCH 4 DESCRIBED IN SECTION 8.4.3 AND OMIT THE USER ATTENTION TEST DESCRIBED IN SECTION 8.1.6. NOTE THIS QUESTION ONLY APPLIES TO REVISION "B" DR70 INTERFACE TESTING.

(3) AT3 ATTN DISABLED [SW5 "OFF"] (L) N ?

A "Y" RESPONSE TO THIS QUESTION WILL BOTH ADD THE OPTIONAL TEST FOR SWITCH 5 DESCRIBED IN SECTION 8.4.4, AND OMIT THE USER ATTENTION TEST DESCRIBED IN SECTION 8.1.6. NOTE THIS QUESTION ONLY APPLIES TO REVISION "B" DR70 INTERFACE TESTING.

(4) AT3 TRANSFER ABORT DISABLED [SW6 "OFF"] (L) N ?

A "Y" RESPONSE TO THIS QUESTION WILL ADD THE OPTIONAL TEST FOR SWITCH 6 DESCRIBED IN SECTION 8.4.5. NOTE THIS QUESTION ONLY APPLIES TO REVISION "B" DR70 INTERFACE TESTING.

(5) BYTE MODE OPERATION [SW7 "ON"] (L) N ?

THIS QUESTION WILL ALLOW THE OPERATOR TO MAKE ALL INTERFACE DATA TRANSFER TESTING IN BYTE MODE. REFER TO THE BYTE MODE TESTS DESCRIBED IN SECTION 8.3.

(6) INPUT PARITY DISABLED [SW8 "OFF"] (L) N ?

THIS QUESTION WILL DISABLE THE PARITY GENERATION TEST DESCRIBED IN SECTION 8.4.2. NOTE THAT THIS QUESTION IS USEFUL ONLY FOR DR70 "B" BOARDS WITHOUT THE INVERT PARITY LOGIC ECO.

(7) OPERATOR SPECIFIED DATA PATTERN (L) N ?

THIS QUESTION WILL ENABLE THE OPERATOR TO SPECIFIY DATA PATTERN AND TRANSFER BLOCK SIZE FOR EITHER BYTE OR WORD MODE TESTS DESCRIBED IN SECTIONS 8.4.6 AND 8.4.7.

(8) ADJUST TRANSFER BANDWIDTH (L) N ?

THIS QUESTION WILL ALLOW THE OPERATOR TO MAKE ADJUSTMENTS TO THE SCLK TRANSFER PERIOD. A "Y" RESPONSE WILL CAUSE THE TEST IN 8.4.8 TO BE EXECUTED BEFORE THE END OF PASS.

4.6 OPERATOR HELP INFORMATION

TWO HELP FILES HAVE BEEN CREATED TO AID THE OPERATOR IN RUNNING THE PROGRAM. THE FILES CONTAIN THE MOST USEFUL INFORMATION CONTAINED IN THIS DOCUMENT. NOTE THE OPERATOR IS ASKED IF HE WANTS THE INFORMATION DISPLAYED ONLY ONCE EACH TIME THE PROGRAM IS LOADED, AFTER ALL SOFTWARE QUESTIONS HAVE BEEN ANSWERED. THE DEFAULT RESPONSE TO EACH QUESTION IS NEGATIVE.

PRINT SUPERVISOR HELP FILE: DRS.HLP (L) N ?

PRINT DIAGNOSTIC HELP FILE: ZDRMA0.HLP (L) N ?

4.7 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THIS PROGRAM IS DESIGNED TO TEST ONLY ONE DR70 INTERFACE UNIT (0-7).

4.8 SAMPLE PROGRAM EXECUTION

THE FOLLOWING SAMPLE PROGRAM EXECUTION WILL ILLUSTRATE THE MINIMUM REQUIRED COMMANDS NECESSARY TO EXECUTE THE DEFAULT TESTS. NOTE THAT THIS EXAMPLE CAN BE USED ONLY IF ALL DEFAULT HARDWARE AND SOFTWARE CONDITIONS HAVE BEEN MET. OPERATOR INPUT IS DENOTED BY THE UNDERLINED TEXT. EXECUTION TIME FOR A SINGLE PASS IS APPROXIMATELY ONE SECOND.

```
.RUN ZDRMA0 <CR>
 --------------
ZDRMA0.BIN

DIAG. RUN-TIME SERVICES

ZDRM-A-0

DR70 REPAIR DIAGNOSTIC

UNIT IS DR70 MASSBUS INTERFACE

RESTART ADDR: 146730

DR>START/PASS:1 <CR>
   ----------------
CHANGE HW (L)   ? N <CR>
                ------
CHANGE SW (L)   ? N <CR>
                ------
PRINT SUPERVISOR HELP FILE:  DRS.HLP (L) N ? N <CR>
                                           ------
PRINT DIAGNOSTIC HELP FILE:  ZDRMA0.HLP (L) N ? N <CR>
                                             ------
ZDRM EOP    1
    0 CUMULATIVE ERRORS

DR>
```

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 4.3 AND 4.5.

4.9 SW1 DEFAULT SWITCH POSITIONS

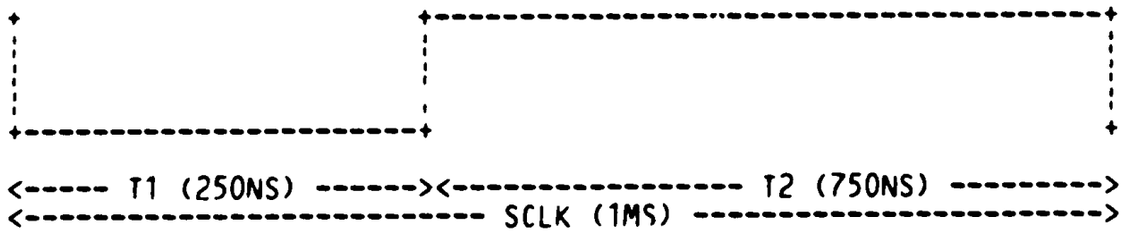
THE FOLLOWING TABLE SHOWS THE NECESSARY SWITCH POSITIONS REQUIRED FOR EXECUTION OF THE DEFAULT TESTS.

SWITCH(ES)	DESCRIPTION	DEFAULT POSITION
SW1-1 SW1-2 SW1-3	MASSBUS DRIVE SELECT BITS 1-3 (SW1-1 IS LSB)	ALL THREE SWITCHES MUST BE CLOSED (ON) TO TEST DRIVE ZERO
SW1-4	AT0 USER ATTENTION ENABLE	AT0 ATTN. ENABLED (ON)
SW1-5	AT3 USER ATTENTION ENABLE	AT3 ATTN. ENABLED (ON)
SW1-6	AT3 TRANSFER ABORT ENABLE	AT3 ABORT ENABLED (ON)
SW1-7	DR70 BYTE MODE OPERATION	WORD MODE XFERS (OFF)
SW1-8	USER INPUT PARITY ENABLE	PARITY ENABLED (ON)

4.10 TRANSFER RATE ADJUSTMENT

THE DR70 MASSBUS DATA TRANSFER RATE IS CONTROLLED BY AN ON-BOARD SYNC CLOCK (SCLK) PULSE. THIS TRANSFER RATE HAS BEEN FACTORY SET AT 1 MHZ BEFORE SHIPMENT. THIS RATE MAY BE SET TOO FAST FOR SOME SYSTEMS, PRODUCING DATA LATE (DLT) ERRORS AT THE MASSBUS CONTROLLER. IN THIS CASE, THE DR70 SCLK SHOULD BE ADJUSTED BY EXECUTING TEST 27 (BANDWIDTH ADJUSTMENT) AND PLACING AN OCILLISCOPE PROBE ON THE M8440 BOARD TEST POINT (TP1). THE FRENQUENCY ADJUSTMENT IS MADE USING THE TWO CALIBRATION POTIENTIOMETERS, R57 AND R32. NOTE THAT R57 CONTROLS THE SCLK LOW PULSE WIDTH (T1) AND SHOULD BE SET AT NO LESS THAN 250 NS FOR PROPER OPERATION. THIS DIAGNOSTIC WILL PASS ALL DATA TRANSFER TESTS WITH SCLK SET AT A MAXIMUM RATE OF 1.12 MHZ ON BOTH RH70 AND RH11 MASSBUS CONTROLLERS.

DR70 SYNC CLOCK (SCLK) PULSE WITH MINIMUM TIMES.



5.0 DEVICE INFORMATION TABLES

THE DR70 P-TABLE ENTRY CONTAINS ONLY THE MINIMUM INFORMATION REQUIRED TO ADDRESS THE DEVICE AND SERVICE INTERRUPTS. NOTE THAT THESE PARAMETERS ARE DETERMINED BY JUMPERS ON THE RH11 OR RH70 MASSBUS CONTROLLER ADDRESS BOARD. REFER TO SECTION 4.4 FOR THE HARDWARE PARAMETER QUESTIONS AND DEFAULT VALUES.

DEVICE ADDRESS: THIS PARAMETER SPECIFIES THE UNIBUS BASE ADDRESS FOR THE MASSBUS CONTROLLER. ALL REGISTERS ARE MAPPED ONTO THE BUS STARTING AT THIS ADDRESS.

VECTOR ADDRESS: THIS PARAMETER SPECIFIES THE INTERRUPT SERVICE ADDRESS VECTOR FOR THE DEVICE, WHICH MUST CONTAIN THE SERVICE ROUTINE START ADDRESS.

PRIORITY: THIS PARAMETER SPECIFIES THE DEVICE INTERRUPT PRIORITY, WHICH DETERMINES AT WHAT PRIORITY THE SERVICE ROUTINE WILL RUN.

DRIVE: THIS PARAMETER CORRESPONDES TO THE DRIVE OR UNIT SELECT SWITCHES SET ON THE INTERFACE.

6.0 ERROR INFORMATION

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 4.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
ERROR MESSAGE
```

NAME = DIAGNOSTIC NAME
TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
NUMBER = ERROR NUMBER
UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBR" FLAGS ARE SET (SECTION 4.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBR" OR "IXR" FLAGS ARE SET (SECTION 4.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

6.1 BASIC ERROR MESSAGES

ALL BASIC ERROR MESSAGES ARE DESIGNED TO BE SPECIFIC AND SELF-EXPLANATORY. PLEASE REFER TO THE DIAGNOSTIC PROGRAM LISTING GLOBAL TEXT SECTION FOR A COMPLETE LIST OF THE BASIC ERROR MESSAGES.

6.1.1 REGISTER VALUE/ADDRESSING ERRORS

''REGISTER ADDRESS ERROR''
''NON-EXISTENT DRIVE ERROR''
''REGISTER INITIALIZATION ERROR''
''REGISTER ERROR''
''CS1 FUNCTION BIT ERROR''
''FUNCTION STATUS BIT ERROR''

6.1.2 MASSBUS CONTROLLER ERRORS

''MASSBUS CONTROLLER ERROR''
''MASSBUS CONTROLLER RDY BIT NOT RESET BY IRY''
''MASSBUS CONTROLLER RDY BIT NOT SET BY IRY''
''CONTROLLER NOT CLEARED BY MASSBUS INIT''

6.1.3 INTERFACE STATUS REGISTER ERRORS

''INTERFACE STATUS ERROR''
''INTERFACE STATUS ERROR ON INTERFACE CLEAR CMD''
''INTERFACE STATUS USER ATTN BIT NOT SET/RESET''
''INTERFACE STATUS USER ATTENTION BIT NOT RESET''
''INTERFACE STATUS IRY NOT SET ON ERROR''
''INTERFACE STATUS IRY BIT RESET WHEN CYC BIT SET''
''INTERFACE STATUS IRY NOT RESET BY DATA TRANSFER CMD''
''IRY NOT SET BY WORD TRANSFER END OF BLOCK''
''IRY NOT SET BY WORD TRANSFER ABORT''
''IRY NOT SET BY INTERFACE CLEAR CMD''
''IRY NOT SET BY BYTE TRANSFER END OF BLOCK''
''IRY NOT SET BY BYTE TRANSFER ABORT''

6.1.4 USER ATTENTION CONDITION ERRORS

```
''ATA NOT SET ON USER ATTN CONDITION''
''ATA NOT SET BY ERR ACTIVE''
''ATA NOT RESET BY INTERFACE CLEAR CMD''
''NO INTERRUPT WHEN INTERFACE STATUS ATA SET''
''ATA NOT RESET BY WRITING TO ATTN SUMMARY BIT''
''ATA NOT RESET BY LOADING DATA TRANSFER CMD''
''ATTENTION SUMMARY BIT NOT SET BY UN.T UNDER TEST''
```

6.1.5 USER INTERFACE ERRORS

```
''ERO NOT SET THROUGH M8432 TEST MODULE''
''ERO NOT RESET BY INTERFACE CLEAR CMD''
''ERR NOT SET BY ERO ACTIVE''
''ERR NOT RESET BY INTERFACE CLEAR CMD''
```

6.1.6 DATA TRANSFER ERRORS

```
''WORD TRANSFER NOT COMPLETE''
''WORD TRANSFER ERROR ON INTERFACE WRITE''
''WORD TRANSFER ERROR ON INTERFACE READ''
''WORD TRANSFER COMPARISON ERROR''
''BYTE TRANSFER NOT COMPLETE''
''BYTE TRANSFER ERROR ON INTERFACE WRITE''
''BYTE TRANSFER ERROR ON INTERFACE READ''
''BYTE TRANSFER COMPARISON ERROR''
```

6.1.7 TRANSFER INTERRUPT ERRORS

```
''NO INTERRUPT ON WORD TRANSFER END OF BLOCK''
''NO INTERRUPT ON BYTE TRANSFER END OF BLOCK''
```

6.1.8 TRANSFER ABORT ERRORS

```
''NO WORD TRANSFER ABORT ON INTERFACE WRITE''
''NO WORD TRANSFER ABORT ON INTERFACE READ''
''NO BYTE TRANSFER ABORT ON INTERFACE WRITE''
''NO BYTE TRANSFER ABORT ON INTERFACE READ''
''CONTROLLER TRE BIT NOT SET BY TRANSFER ABORT LOGIC''
```

6.1.9 CONTROL/DATA PARITY ERRORS

''INTERFACE STATUS IDP NOT SET BY DATA PARITY ERROR''
''INTERFACE STATUS IDP NOT RESET BY INTERFACE CLEAR CMD''
''INTERFACE STATUS ICP NOT RESET BY WRITE TO IDP BIT''
''INTERFACE STATUS ICP NOT SET BY CONTROL PARITY ERROR''
''INTERFACE STATUS ICP NOT RESET BY INTERFACE CLEAR CMD''
''INTERFACE STATUS ICP NOT RESET BY WRITE TO ICP BIT''

6.1.10 USER ATTENTION/ABORT SWITCH ERRORS

''SW4 DOES NOT DISABLE AT0 INPUT TO INTERFACE ATA''
''SW5 DOES NOT DISABLE AT3 INPUT TO INTERFACE ATA''
''SW6 DOES NOT DISABLE AT3 TRANSFER ABORT LOGIC''

6.2 EXTENDED ERROR MESSAGES

EACH ERROR CONDITION SENSED WHILE TESTING THE INTERFACE WILL PRODUCE AT LEAST ONE EXTENTED ERROR MESSAGE IN ADDITION TO THE ERROR BANNER AND BASIC MESSAGE. THESE EXTENDED MESSAGES WILL PROVIDE USEFUL INFORMATION ABOUT THE INTERFACE/CONTROLLER REGISTER STATUS. EACH TYPE OF EXTENDED IS DISCUSSED BELOW.

6.2.1 REGISTER ADDRESSING ERRORS

ALL REGISTER ADDRESSING ERRORS ARE ACCUMULATED AND REPORTED IN TABULAR FORM. AN EXAMPLE IS GIVEN BELOW.

```
REG     ADRS
DR.CS1  172440
DI .IS  172452
```

6.2.2 REGISTER INITIALIZATION ERRORS

ONLY SPECIFIC REGISTER INITIALIZATION VALUES ARE TESTED BY THE PROGRAM. ANY UNEXPECTED VALUES ARE ACCUMULATED AND LISTED IN TABULAR FORM, WITH THEIR ACTUAL, EXPECTED, XOR AND EXPANDED BIT VALUES. AN EXAMPLE IS GIVEN BELOW.

```
REG     ACT     EXP     XOR     BIT(S)
DR.CS1  144260  004270  140010  SC TRE F3
```

6.2.3 REGISTER ERRORS

ALL SPECIFIC REGISTER ERRORS ARE REPORTED WITH THEIR ACTUAL, EXPECTED, XOR AND EXPANDED BIT VALUES. AN EXAMPLE IS GIVEN BELOW.

```
REG      ACT     EXP     XOR     BIT(S)
RH.CS1   100700  000300  100400  DLT MDPE
```

6.2.4 CONTROLLER/INTERFACE STATUS ERRORS

IN SOME CASES, A SPECIFIC REGISTER ERROR MAY HAVE BEEN CAUSED BY THE INTERACTION BETWEEN THE MASSBUS CONTROLLER AND THE INTERFACE, AND A SINGLE REGISTER VALUE DISPLAY IS INSUFFIENT TO ISOLATE THE PROBLEM. IN THESE CASES, ALL CONTROLLER AND INTERFACE REGISTERS ARE DISPLAYED WITH THE INTENT TO GIVE THE OPERATOR A SNAPSHOT OF THE SYSTEM FOR ANALYSIS. AN EXAMPLE IS GIVEN BELOW.

```
CONTROLLER STATUS
RH.CS1          [172440]  144260  ;SC TRE DVA RDY F4 F3
RH.WC           [172442]  177664  ;
RH.BA           [172444]  003224  ;
RH.CS2          [172450]  100307  ;DLT OR IR U2 U1 U0

INTERFACE STATUS
DR.FS           [172446]  004010  ;FR3 DS3
DR.IS           [172452]  100210  ;ATA IRY AT3
DR.IB           [172454]  000400  ;
DR.AS           [172456]  000200  ;AS7
DR.OB           [172460]  000000  ;
DR.DT           [172466]  000005  ;
```

6.2.5 DATA COMPARISON ERRORS

ALL DATA TRANSFER TESTS WILL COMPARE DATA READ AGAINST DATA WRITTEN. ANY DISCREPANCES ARE RECORDED AND ONLY THE FIRST EIGHT OCCURENCES ARE LISTED, ALONG WITH THE TOTAL ERROR COUNT. IN ADDITION, THE INPUT BUFFER DATA ADDRESS AND DECIMAL WORD NUMBER ARE GIVEN, TO AID IN ISOLATION OF PARTICULAR DATA TRANSFER PROBLEMS. AN EXAMPLE IS GIVEN BELOW.

```
ADRS    WRD     RECVD   EXPTD   XOR
002040    1     107776  177776  070000
002042    2     117777  177775  060002
002056   10     136777  176777  040000

TOTAL ERRORS:    3
```

6.2.6 DATA PATTERN DISPLAY

DURING INVERTED PARITY GENERATION TESTING, THE PROGRAM WILL DISPLAY THE STATUS OF ALL REGISTERS, AS DESCRIBED ABOVE. IN ADDITION, THE

PROGRAM WILL DISPLAY THE SPECIFIC DATA PATTERN THAT PRODUCED THE
ERROR IN THAT SUBTEST, AS SHOWN.

DATA PATTERN: 177777

6.3 FORCED MESSAGES

DETECTION OF CERTAIN ERRORS INDICATE A CRITICAL INSTALLATION OR HARDWARE CONFIGURATION FAULT AND WILL FORCE THE APPROPRIATE MESSAGE TO BE DISPLAYED TO FACILITATE CORRECTION OF THE PROBLEM. NOTE THAT THE MESSAGE(S) ARE DISPLAYED REGARDLESS OF OPERATOR SPECIFIED FLAGS/ SWITCHES AND WILL ONLY BE LISTED ONCE EASH TIME THE PROGRAM IS LOADED.

''P-TABLE ERROR ON UNIT 0''
''CHECK MASSBUS CONTROLLER CSR ADDRESS''
''CHECK DRIVE SELECT SWITCHES 1-3''
''CHECK UNIT FOR POWER ON''
''CHECK SWITCH SW1-7 SET TO ''OFF'' POSITION''
''CHECK SWITCH SW1-8 SET TO ''ON'' POSITION''
''CHECK M8432 TEST MODULE LOOP-BACK CABLE''

7.0 PERFORMANCE AND PROGRESS REPORTS

AT THE END OF EACH PASS, THE PASS COUNT IS GIVEN ALONG WITH THE TOTAL NUMBER OF ERRORS REPORTED SINCE THE DIAGNOSTIC WAS STARTED. THE "EOP" SWITCH CAN BE USED TO CONTROL HOW OFTEN THE END OF PASS MESSAGE IS PRINTED. SECTION 4.2 DESCRIBES SWITCHES.

ADDITIONALLY THE PROGRAM SUPPORTS THE "PRINT" FUNCTION, WHICH ALLOWS THE OPERATOR TO DISPLAY STATISTICAL DATA ABOUT THE TESTS. EACH TEST THAT COMPLETES EXECUTION WILL LOG ITS PASS AND ERROR COUNT INTO THE REPORT TABLES. EACH TIME THE PROGRAM INITIALIZES THROUGH THE START COMMAND CODE, ALL REPORT TABLE ENTRIES ARE CLEARED. THE PROGRAM WILL CONTINUE TO LOG THE INFORMATION UNTIL THE REQUIRED NUMBER OF PASSES ARE EXECUTED, OR AN OPERATOR BREAK IS DETECTED. THE OPERATOR MAY THEN USE THE PRINT COMMAND TO DISPLAY THE TEST SUMMARY, AS SHOWN BELOW. NOTE THAT DATA IS DISPLAYED ONLY FOR TESTS THAT WERE EXECUTED.

```
DR>STA/TES:1-4:17-20/PAS:1000/EOP:1000/FLA:IER

CHANGE HW ? N

CHANGE SW ? N
ZDRMA0 EOP 1000
         160 CUMULITIVE ERRORS
DR>PRI

DIAGNOSTIC SUMMARY:

TEST  PASSES  ERRORS
 1      1000       0
 2      1000       0
 3      1000       0
 4      1000      12
17      1000     100
18      1000       0
19      1000       0
20      1000      48
```

7.1 REPORT DATA CONTROL

EACH TIME THE "PRINT" COMMAND IS GIVEN, THE REPORT TABLE DATA IS RESET TO ZERO FOR ALL TESTS. THIS IS ALSO TRUE FOR BOTH "START" AND "RESTART" COMMANDS. NOTE THE "CONTINUE" COMMAND CAN BE USED TO AVOID A TABLE RESET.

8.0 PROGRAM DESCRIPTION

THE MAIN GROUP OF TESTS IN THE PROGRAM ARE CONSIDERED THE DEFAULT TEST GROUP, WHICH WILL TEST THE BASIC INTERFACE FUNCTIONS. THE INTERFACE IS DESIGNED TO ACCOMODATE SEVERAL ADDITIONAL FUNCTIONS, WHICH MAY BE SELECTED BY THE USER, USING THE ON-BOARD DIP SWITCHES.

THESE FUNCTIONS ARE INCLUDED IN THE TEST SEQUENCE, ONLY WHEN SELECTED USING THE PROGRAM SOFTWARE PARAMETERS, AND MAY AUTOMATICALLY EXCLUDE SOME DEFAULT TESTS AFFECTED BY THE OPTIONAL FUNCTIONS. IT IS ASSUMED THAT THE OPERATOR IS FAMILIAR WITH THE REQUIREMENTS OF THE INTERFACE ENVIRONMENT AND THE TESTS DESCRIBED BELOW BEFORE THE PROGRAM IS LOADED.

8.1 DEFAULT REGISTER LOGIC TESTS

SECTIONS 8.1.1 - 8.1.11 DESCRIBE ALL DEFAULT TESTS AND THEIR CONDITIONS FOR EXECUTION. WITH THE EXCEPTION OF THE REGISTER ADDRESSING TEST, ALL TESTS BEGIN WITH A MASSBUS CONTROLLER CLEAR, WHICH WILL INITIALIZE ALL REGISTERS TO A KNOWN STATE. NOTE THAT ANY ERROR DETECTED AFTER A MASSBUS CONTROLLER CLEAR IS CONSIDERED DEVICE FATAL AND WILL DISPLAY ALL REGISTERS CONTENTS AND ABORT THE CURRENT PASS.

8.1.1 REGISTER ADDRESSING: (TEST 1)

THIS TEST WILL DETERMINE THAT THE RH11 OR RH70 MASSBUS CONTROLLER IS CONFIGURED WITH THE CORRECT UNIBUS DEVICE ADDRESS, AND THAT THE CORRECT DR70 INTERFACE UNIT IS UNDER TEST. TEST FAILURE INDICATES THAT EITHER THE SPECIFIED REGISTER(S) CANNOT BE ADDRESSED, AND THAT THE MASSBUS CONTROLLER ADDRESS IS INCORRECT, OR THE INCORRECT UNIT HAS BEEN SELECTED AND THE INTERFACE UNIT SELECT SWITCHES SHOULD BE CHECKED. NOTE THAT TEST FAILURE WILL ABORT THE ENTIRE PASS.

8.1.2 REGISTER INITIALIZATION: (TEST 2)

THIS TEST WILL CHECK THAT SPECIFIC REGISTERS HAVE BEEN CORRECTLY INITIALIZED BY A MASSBUS CONTROLLER CLEAR SIGNAL. TEST FAILURE WILL LIST THE REGISTER(S) WITH THEIR ACTUAL, EXPECTED AND XOR VALUES AFTER THE MASSBUS CLEAR. NOTE THAT NOT ALL REGISTERS ARE TESTED, AND THAT FAILURE MAY INDICATE THAT EITHER THE CONTROLLER OR THE INTERFACE ARE BAD OR THAT THE INCORRECT DR70 REVISION HAS BEEN SELECTED.

8.1.3 CONTROL/STATUS 1 REGISTER FUNCTION BITS: (TEST 3)

THIS TEST WILL DETERMINE THAT ALL CONTROL/STATUS 1 REGISTER FUNCTION BITS MAY BE SET/RESET THROUGH THE MASSBUS CONTROLLER. TEST FAILURE WILL LIST THE CS1 REGISTER ACTUAL, EXPECTED AND XOR VALUES AND INDICATES PROBABLE FAULT ON THE INTERFACE BOARD.

8.1.4 INTERFACE FUNCTION/STATUS REGISTER BITS: (TEST 4)

THIS TEST WILL DETERMINE THAT ALL INTERFACE STATUS DRIVE STATUS BITS MAY BE SET/RESET BY THE USER INTERFACE. TEST FAILURE WILL LIST THE FUNCTION/STATUS REGISTER ACTUAL, EXPECTED AND XOR VALUES, AND MAY INDICATE A FAULT ON THE INTERFACE BOARD. NOTE THAT THIS TEST DEPENDS UPON PROPER OPERATION OF THE M8432 TEST MODULE AND LOOP-BACK CABLING. IN THE EVENT THAT NO DRIVE STATUS BITS MAY BE SET, THE OPERATOR IS PROMPTED TO CHECK THAT THE LOOP-BACK CABLE HAS BEEN PROPERLY INSTALLED.

8.1.5 INTERFACE STATUS ATTENTION CONDITION: (TEST 5)

THIS TEST WILL BE EXECUTED ONLY WHEN THE DEFAULT USER ATTENTION FUNCTIONS ARE IN EFFECT. THE TEST WILL VERIFY THAT THE INTERFACE STATUS REGISTER USER ATTENTION BITS MAY BE SET/RESET BY THE DRIVE UNDER TEST. TEST FAILURE WILL LIST THE INTERFACE STATUS REGISTER ACTUAL, EXPECTED AND XOR VALUES, AND AGAIN MAY INDICATE A PROBLEM IN USING THE M8432 TEST MODULE AND CABLE.

8.1.6 INTERFACE STATUS USER ATTENTION CONDITIONS: (TEST 6)

THIS TEST WILL BE EXECUTED ONLY WHEN THE DEFAULT USER ATTENTION FUNCTIONS ARE IN EFFECT. THE TEST WILL VERIFY THAT THE INTERFACE STATUS REGISTER ATTENTION BIT IS SET BY USER ATTENTION CONDITIONS AND MAY BE RESET BY A DRIVE CLEAR COMMAND. TEST FAILURE WILL LIST THE INTERFACE STATUS REGISTER ACTUAL, EXPECTED AND XOR VALUES, AND INDICATES A FAULT ON THE INTERFACE BOARD.

8.1.7 INTERFACE STATUS USER ERROR CONDITION: (TEST 7)

THIS TEST CHECKS FIRST THAT THE INTERFACE STATUS REGISTER USER ERROR BIT MAY BE SET, AND WILL SET BOTH THE GENERAL ERROR AND ATTENTION CONDITION BITS, AND SECONDLY THAT ALL BITS MAY BE RESET BY A DRIVE CLEAR COMMAND. TEST FAILURE WILL LIST THE INTERFACE STATUS REGISTER ACTUAL, EXPECTED AND XOR VALUES, AND INDICATES EITHER A FAULT ON THE INTERFACE BOARD OR A LOOP-BACK CABLE PROBLEM.

8.1.8 INTERFACE STATUS READY STATUS ON ERROR: (TEST 8)

THIS TEST WILL VERIFY THAT THE INTERFACE READY STATUS WILL BE RESET BY LOADING A VALID DATA TRANSFER COMMAND INTO THE DRIVE CONTROL REGISTER, AND WILL BE SET BY A USER ERROR CONDITION. TEST FAILURE WILL DISPLAY BOTH THE CONTROLLER AND INTERFACE REGISTERS CONTENTS.

8.1.9 INTERFACE STATUS ADVANCED ATTENTION FUNCTIONS: (TEST 9)

THIS TEST WILL VERIFY THAT AN INTERFACE ATTENTION CONDITION WILL GENERATE A DEVICE INTERRUPT, AND THAT UNIT UNDER TEST WILL SET THE CORRESPONDING ATTENTION SUMMARY REGISTER BIT. IN ADDITION, THE TEST WILL INSURE THAT THE ATTENTION CONDITION IS REMOVED BY WRITING TO THE ATTENTION SUMMARY REGISTER BIT. FAILURE TO RECEIVE INTERRUPT WILL DISPLAY ALL REGISTERS CONTENTS, AND MAY INDICATE THAT THE INCORRECT INTERRUPT VECTOR ADDRESS OR PRIORITY HAS BEEN SELECTED FOR THE MASSBUS CONTROLLER. AN ATTENTION SUMMARY REGISTER FAILURE WILL LIST THE REGISTER ACTUAL, EXPECTED AND XOR VALUES, AND INDICATES A FAULT ON THE INTERFACE BOARD.

8.2 DEFAULT WORD TRANSFER TESTS

SECTIONS 8.2.1 - 8.2.5 DESCRIBE ALL INTERFACE WORD TRANSFER TESTS. THESE TESTS ARE EXECUTED WITH THE DEFAULT TESTS, UNLESS BYTE TRANSFERS ARE SELECTED BY THE OPERATOR. THE TESTS WILL TRANSFER 64-WORD BLOCKS TO AND FROM THE INTERFACE, UTILIZING THE TEST MODULE SILO AND CABLING.

8.2.1 WORD DATA TRANSFER LOGIC: (TEST 10)

THIS TEST WILL BE EXECUTED ONLY UNDER THE DEFAULT (WORD) TRANSFER MODE CONDITIONS. THE TEST WILL CHECK THAT AN INTERFACE CYCLE REQUEST WILL NOT RESET EITHER THE INTERFACE OR CONTROLLER READY STATUS BITS UNTIL A VALID DATA TRANSFER COMMAND IS ISSUED TO THE DRIVE, AND THAT THE READY STATUS IS PRESENT ON TRANSFER COMPLETE. TEST FAILURE WILL DISPLAY ALL REGISTERS CONTENTS, AND INDICATES A PROBABLE FAULT ON THE INTERFACE BOARD.

8.2.2 WORD DATA TRANSFER INTERRUPTS: (TEST 11)

THIS TEST WILL BE EXECUTED ONLY UNDER THE DEFAULT (WORD) TRANSFER MODE CONDITIONS. THE TEST WILL VERIFY THAT BOTH BLOCK WRITE AND BLOCK READ TO AND FROM THE INTERFACE WILL GENERATE AN INTERRUPT. BOTH INTERFACE AND CONTROLLER STATUS ARE CHECKED FOR ERRORS, AND AN INTERRUPT MUST BE RECEIVED. ANY ERROR CONDITIONS WILL CAUSE TEST FAILURE AND DISPLAY OF ALL REGISTERS CONTENTS.

8.2.3 WORD DATA TRANSFER INTEGRITY: (TEST 12)

THIS TEST WILL BE EXECUTED ONLY UNDER THE DEFAULT (WORD) TRANSFER MODE CONDITIONS. THE TEST WILL PERFORM BLOCK TRANSFER TO AND FROM THE INTERFACE VIA THE TEST MODULE, AND TEST THE DATA INTEGRITY. FOUR DISTINCT DATA PATTERNS ARE USED, TO ISOLATE SPECIFIC REGISTER OR BUS FAULTS. TEST FAILURE MAY BE DUE TO CONTROLLER/INTERFACE ERROR STATUS, NO INTERRUPT RECEIVED OR ERROR IN COMPARISON OF DATA READ AND DATA WRITTEN. STATUS/INTERRUPT ERRORS WILL CAUSE ALL REGISTERS TO BE DISPLAYED. DATA ERRORS WILL CAUSE THE RECEIVED, EXPECTED AND XOR DATA VALUES TO BE DISPLAYED. ONLY THE FIRST EIGHT DATA ERROR VALUES ARE LISTED ALONG WITH THE TOTAL ERROR COUNT.

8.2.4 WORD TRANSFER ABORT THROUGH XABT BIT: (TEST 13)

THIS TEST WILL BE EXECUTED ONLY UNDER THE DEFAULT (WORD) TRANSFER CONDITIONS AND WILL VERIFY THAT THE REVISION "B" BOARD DATA TRANSFERS MAY BE TERMINATED BY WRITING TO THE INTERFACE STATUS REGISTER XABT BIT. BOTH READ AND WRITE TRANSFERS ARE TESTED, AND THE TRANSFER MUST HALT ONLY A FEW CYCLES AFTER THE XABT BIT IS SET. TEST FAILURE WILL CAUSE ALL REGISTERS CONTENTS TO BE DISPLAYED, AND MAY INDICATE THE INCORRECT DR70 REVISION HAS BEEN SPECIFIED.

8.2.5 WORD TRANSFER ABORT THROUGH USER ATTENTION: (TEST 14)

THIS TEST WILL BE EXECUTED ONLY UNDER THE DEFAULT (WORD) TRANSFER AND DEFAULT USER ATTENTION TEST CONDITIONS. THE TEST WILL VERIFY THAT THE REVISION "B" BOARD DATA TRANSFERS MAY BE TERMINATED BY SETTING THE USER ATTENTION CONDITION AT03. BOTH READ AND WRITE TRANSFERS ARE TESTED, AND THE TRANSFER MUST HALT WITHIN A FEW CYCLES AFTER THE INTERFACE STATUS REGISTER AT03 ATTENTION BIT IS SET. TEST FAILURE MAY OCCUR IF THE TRANSFER DOES NOT HALT SOON ENOUGH OR IF THE AT03 ATTENTION ABORT ENABLE (SWITCH 6) HAS BEEN SET INCORRECTLY, AND WILL CAUSE THE CONTENTS OF ALL REGISTERS TO BE DISPLAYED.

8.3 BYTE TRANSFER TESTS

SECTIONS 8.3.1 - 8.3.5 DESCRIBE THE INTERFACE BYTE TRANSFER TESTS. THESE TESTS WILL EXERCISE THE INTERFACE BYTE TIMING CONTROL SECTION OF THE INTERFACE, AND ARE EXECUTED ONLY WHEN SELECTED BY THE OPERATOR. THE TESTS WILL TRANSFER 64 PACKED-BYTE BLOCKS TO AND FROM THE INTERFACE UTILIZING THE TEST MODULE SILO AND CABLING. NOTE THAT BYTE MODE TESTS CANNOT IMPLICITLY DETECT WHETHER THE DR70 IS OPERATING IN WORD OR BYTE TRANSFER MODES.

8.3.1 BYTE DATA TRANSFER LOGIC: (TEST 15)

THIS TEST WILL BE EXECUTED ONLY WHEN THE OPERATOR SELECTS THE BYTE TRANSFER MODE CONDITIONS. THE TEST WILL CHECK THAT AN INTERFACE CYCLE REQUEST WILL NOT RESET EITHER THE INTERFACE OR CONTROLLER READY STATUS BITS UNTIL A VALID DATA TRANSFER COMMAND IS ISSUED TO THE DRIVE, AND THAT THE READY STATUS IS PRESENT ON TRANSFER COMPLETE. TEST FAILURE WILL DISPLAY ALL REGISTERS CONTENTS, AND INDICATES A PROBABLE FAULT ON THE INTERFACE BOARD.

8.3.2 BYTE DATA TRANSFER INTERRUPTS: (TEST 16)

THIS TEST WILL BE EXECUTED ONLY UNDER THE OPTIONAL BYTE TRANSFER MODE CONDITIONS. THE TEST WILL VERIFY THAT BOTH BLOCK WRITE AND BLOCK READ TO AND FROM THE INTERFACE WILL GENERATE AN INTERRUPT. BOTH INTERFACE AND CONTROLLER STATUS ARE CHECKED FOR ERRORS, AND AN INTERRUPT MUST BE RECEIVED. ANY ERROR CONDITIONS WILL CAUSE TEST FAILURE AND DISPLAY OF ALL REGISTERS CONTENTS.

8.3.3 BYTE DATA TRANSFER INTEGRITY: (TEST 17)

THIS TEST WILL BE EXECUTED ONLY UNDER THE OPTIONAL BYTE TRANSFER MODE CONDITIONS. THE TEST WILL PERFORM BLOCK TRANSFER TO AND FROM THE INTERFACE VIA THE TEST MODULE, AND TEST THE DATA INTEGRITY. FOUR DISTINCT DATA PATTERNS ARE USED, TO ISOLATE SPECIFIC REGISTER OR BUS FAULTS. TEST FAILURE MAY BE DUE TO CONTROLLER/INTERFACE ERROR STATUS, NO INTERRUPT RECEIVED OR ERROR IN COMPARISON OF DATA READ AND DATA WRITTEN. STATUS/INTERRUPT ERRORS WILL CAUSE ALL REGISTERS TO BE DISPLAYED. DATA ERRORS WILL CAUSE THE RECEIVED, EXPECTED AND XOR DATA VALUES TO BE DISPLAYED. ONLY THE FIRST EIGHT DATA ERROR VALUES ARE LISTED ALONG WITH THE TOTAL ERROR COUNT.

8.3.4 BYTE TRANSFER ABORT THROUGH XABT BIT: (TEST 18)

THIS TEST WILL BE EXECUTED ONLY UNDER THE OPTIONAL BYTE TRANSFER CONDITIONS AND WILL VERIFY THAT THE REVISION 'B" BOARD DATA TRANSFERS MAY BE TERMINATED BY WRITING TO THE INTERFACE STATUS REGISTER XABT BIT. BOTH READ AND WRITE TRANSFERS ARE TESTED, AND THE TRANSFER MUST HALT ONLY A FEW CYCLES AFTER THE XABT BIT IS SET. TEST FAILURE WILL CAUSE ALL REGISTERS CONTENTS TO BE DISPLAYED, AND MAY INDICATE THE INCORRECT DR70 REVISION HAS BEEN SPECIFIED.

8.3.5 BYTE TRANSFER ABORT THROUGH USER ATTENTION: (TEST 19)

THIS TEST WILL BE EXECUTED ONLY UNDER THE OPTIONAL BYTE TRANSFER AND DEFAULT USER ATTENTION TEST CONDITIONS. THE TEST WILL VERIFY THAT THE REVISION 'B" BOARD DATA TRANSFERS MAY BE TERMINATED BY SETTING THE USER ATTENTION CONDITION AT03. BOTH READ AND WRITE TRANSFERS ARE TESTED, AND THE TRANSFER MUST HALT WITHIN A FEW CYCLES AFTER THE INTERFACE STATUS REGISTER AT03 ATTENTION BIT IS SET. TEST FAILURE MAY OCCUR IF THE TRANSFER DOES NOT HALT SOON ENOUGH OR IF THE AT03 ATTENTION ABORT ENABLE (SWITCH 6) HAS BEEN SET INCORRECTLY, AND WILL CAUSE THE CONTENTS OF ALL REGISTERS TO BE DISPLAYED.

8.4 ADVANCED FEATURE TESTS

THE TESTS DESCRIBED IN 8.4.1 - 8.4.2 WILL NOT BE EXECUTED UNDER ANY DEFAULT TEST CONDITIONS. SELECTION OF ONE OR MORE OPTIONAL TESTS CAN ONLY BE MADE IN RESPONSE TO THE SOFTWARE TABLE PARAMETER SECTION. NOTE THAT WHEN LOADED, ALL SOFTWARE TABLE OPTIONAL PARAMETERS ARE CLEARED, BUT ONCE SELECTED WILL BECOME THE DEFAULT VALUES UNLESS RESET THROUGH ANOTHER SOFTWARE TABLE PARAMETER DIALOGUE (SECTION 4.5).

NOTE THAT THE DIAGNOSTIC CANNOT IMPLICITLY DETECT ANY SWITCH POSITIONS AND MUST RELY UPON ACCURACY OF ANSWERS GIVEN TO THE SOFTWARE QUESTIONS TO DETERMINE MODES OF OPERATION. EXTRANEOUS ERRORS MAY OCCUR WHEN ANY SWITCH POSITION IS CHANGED WITHOUT RESTARTING THE PROGRAM AND CHANGING THE RESPONSE TO THE APPROPRIATE SOFTWARE QUESTION.

8.4.1 INTERFACE STATUS CONTROL PARITY ERROR: (TEST 20)

THIS TEST IS EXECUTED UNDER UNDER THE DEFAULT DR70 REVISION LEVEL. THE TEST WILL VERIFY THAT THE INTERFACE STATUS REGISTER CONTROL PARITY ERROR BIT MAY BE SET BY ISSUING A COMMAND WITH INVERTED PARITY, AND WILL BE RESET BY ISSUING AN INTERFACE WRITE COMMAND WITH NORMAL PARITY. TEST FAILURE WILL DISPLAY ALL REGISTERS CONTENTS.

8.4.2 INTERFACE STATUS DATA PARITY ERROR: (TEST 21)

THIS TEST WILL BE EXECUTED UNDER THE DEFAULT DR70 REVISION LEVEL. THE TEST WILL DETERMINE THAT THE INTERFACE STATUS REGISTER DATA PARITY ERROR BIT MAY BE SET BY INVERTED DATA PARITY GENERATION ON THE INTERFACE, AND WILL BE RESET BY ISSUING A DRIVE CLEAR COMMAND TO THE INTERFACE. THE TEST MAY BE EXCLUDED THROUGH OPERATOR SELECTION. TEST FAILURE WILL DISPLAY ALL REGISTERS CONTENTS, AND MAY INDICATE A PARITY GENERATION ERROR, OR THAT THE INCORRECT USER PARITY ENABLE/ INHIBIT HAS BEEN SELECTED.

8.4.3 INTERFACE USER ATTENTION AT0 DISABLE: (TEST 22)

THIS TEST IS EXECUTED ONLY WHEN TESTING THE REVISION 'B'' INTERFACE AND WHEN SELECTED BY THE OPERATOR THROUGH THE SOFTWARE QUESTIONNING. THE TEST WILL VERIFY THAT THE DR70 BOARD SWITCH 4 WILL INHIBIT THE USER ATTENTION AT0 FROM INPUT TO INTERFACE STATUS ATA, WHEN IN THE 'OFF'' POSITION. TEST FAILURE MAY INDICATE THAT THE INTERFACE BOARD SWITCH HAS BEEN SET INCORRECTLY AND WILL DISPLAY THE ACTUAL, EXPECTED, AND XOR INTERFACE STATUS REGISTER VALUES.

8.4.4 INTERFACE USER ATTENTION AT3 DISABLE: (TEST 23)

THIS TEST IS EXECUTED ONLY WHEN TESTING THE REVISION 'B'' INTERFACE AND WHEN SELECTED BY THE OPERATOR THROUGH THE SOFTWARE QUESTIONNING. THE TEST WILL VERIFY THAT THE DR70 BOARD SWITCH 5 WILL INHIBIT THE USER ATTENTION AT3 FROM INPUT TO INTERFACE STATUS ATA, WHEN IN THE 'OFF'' POSITION. TEST FAILURE MAY INDICATE THAT THE INTERFACE BOARD SWITCH HAS BEEN SET INCORRECTLY AND WILL DISPLAY THE ACTUAL, EXPECTED, AND XOR INTERFACE STATUS REGISTER VALUES.

8.4.5 USER ATTENTION AT3 TRANSFER ABORT DISABLE: (TEST 24)

THIS TEST IS EXECUTED ONLY WHEN TESTING THE REVISION 'B'' INTERFACE AND WHEN SELECTED BY THE OPERATOR THROUGH THE SOFTWARE QUESTICNNING. THE TEST WILL VERIFY THAT THE DR70 BOARD SWITCH 6 WILL INHIBIT THE USER ATTENTION AT3 FROM CAUSING A DATA TRANSFER ABORT, WHEN IN THE 'OFF'' POSITION. THE TEST IS MADE USING ONLY READ BLOCK DATA TRANSFERS, AND MAY BE EXECUTED IN EITHER WORD OR BYTE TRANSFER MODE. TEST FAILURE MAY INDICATE THAT THE INTERFACE BOARD SWITCH HAS BEEN SET INCORRECTLY AND WILL DISPLAY THE ACTUAL, EXPECTED, AND XOR INTERFACE STATUS REGISTER VALUES.

8.4.6 WORD BLOCK TRANSFER (OPERATOR DATA PATTERN) (TEST 25)

THIS TEST IS DESIGNED TO AID THE OPERATOR IN ISOLATING PROBLEMS WHICH MAY BE DATA SPECIFIC. THE TEST IS EXEUCTED ONLY BY OPERATOR SELECTION IN WORD TRANSFER MODE. THE OPERATOR MAY SUPPLY A ONE WORD DATA PATTERN AND TRANSFER BLOCK SIZE. THE PROGRAM WILL CONTINUE TO WRITE AND READ BLOCKS TO AND FROM THE INTERFACE, UNTIL OPERATOR INTERVENTION THROUGH A CONTROL/C BREAK. TEST FAILURE INDICATORS ARE IDENTICAL TO THE DEFAULT WORD TRANSFER ERRORS (TEST 12).

8.4.7 BYTE BLOCK TRANSFER (OPERATOR DATA PATTERN) (TEST 26)

THIS TEST IS DESIGNED TO AID THE OPERATOR IN ISOLATING PROBLEMS WHICH MAY BE DATA SPECIFIC. THE TEST IS EXEUCTED ONLY BY OPERATOR SELECTION IN BYTE TRANSFER MODE. THE OPERATOR MAY SUPPLY A ONE BYTE DATA PATTERN AND TRANSFER BLOCK SIZE. THE PROGRAM WILL CONTINUE TO WRITE AND READ BLOCKS TO AND FROM THE INTERFACE, UNTIL OPERATOR INTERVENTION THROUGH A CONTROL/C BREAK. TEST FAILURE INDICATORS ARE IDENTICAL TO THE DEFAULT BYTE TRANSFER ERRORS (TEST 17).

8.4.8 TRANSFER BANDWIDTH ADJUSTMENT: (TEST 27)

THIS TEST WILL OPTIONALLY ALLOW THE OPERATOR TO ADJUST THE TRANSFER BANDWIDTH OR SCLK PERIOD TO MATCH THE SYSTEM REQUIREMENTS OF THE INTERFACE, USING AN OSCILLISCOPE TO MAESURE THIS PERIOD DURING WORD OR BYTE MODE DATA TRANSFERS. THE TEST WILL PROMPT THE OPERATOR TO BEGIN THE ADJUSTMENT AND TO INPUT A CONTROL/C FROM THE CONSOLE DEVICE WHEN THE BANDWIDTH IS SET. THE TEST THEN BEGINS TO LOOP ON BLOCK WRITE TRANSFERS TO THE INTERFACE, TO CONTINUOUSLY PROVIDE AN SCLK PERIOD THAT CAN BE OBSERVED AND ADJUSTED.

PROGRAM HEADER

```
                                .SBTTL  PROGRAM HEADER

                                        .ENABL  ABS,AMA
002000                                  .       =       2000

002000                                  BGNMOD

002000                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;      POINTER TABLE AND HEADER CALL
002000                          STARS
                                ;;*****************************************************************

002000                                  POINTER         BGNSW,BGNSFT,BGNRPT,BGNSETUP

002000                                  HEADER          ZDRM,A,0,0,1
002000                          L$NAME::                ;DIAGNOSTIC NAME
002000  132                             .ASCII /Z/
002001  104                             .ASCII /D/
002002  122                             .ASCII /R/
002003  115                             .ASCII /M/
002004  000                             .BYTE   0
002005  000                             .BYTE   0
002006  000                             .BYTE   0
002007  000                             .BYTE   0
002010                          L$REV::                 ;REVISION LEVEL
002010  101                             .ASCII  /A/
002011                          L$DEPO::                ;0
002011  060                             .ASCII  /0/
002012                          L$UNIT::                ;NUMBER OF UNITS
002012  000001                          .WORD   T$PTHV
002014                          L$TIML::                ;LONGEST TEST TIME
002014  000000                          .WORD   0
002016                          L$HPCP::                ;POINTER TO H.W. QUES.
002016  041132                          .WORD   L$HARD
002020                          L$SPCP::                ;POINTER TO S.W. QUES.
002020  041300                          .WORD   L$SOFT
002022                          L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002214                          .WORD   L$HW
002024                          L$SPTP::                ;PTR. TO S.W. PTABLE
002024  002226                          .WORD   L$SW
002026                          L$LADP::                ;DIAG. END ADDRESS
002026  042064                          .WORD   L$LAST
002030                          L$STA::                 ;RESERVED FOR APT STATS
002030  000000                          .WORD   0
002032                          L$CO::
002032  000000                          .WORD   0
002034                          L$DTYP::                ;DIAGNOSTIC TYPE
002034  000001                          .WORD   1
002036                          L$APT::                 ;APT EXPANSION
002036  000000                          .WORD   0
002040                          L$DTP::                 ;PTR. TO DISPATCH TABLE
002040  002124                          .WORD   L$DISPATCH
002042                          L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000000                          .WORD   0
002044                          L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
```

PROGRAM HEADER

```
002044  000000                          .WORD   0
002046                          L$EXP1::                ;EXPANSION WORD
002046  000000                          .WORD   0
002050                          L$MREV::                ;SVC REV AND EDIT #
002050     003                          .BYTE   C$REVISION
002051     003                          .BYTE   C$EDIT
002052                          L$EF::                  ;DIAG. EVENT FLAGS
002052  000000                          .WORD   0
002054  000000                          .WORD   0
002056                          L$SPC::
002056  000000                          .WORD   0
002060                          L$DEVP::                ; POINTER TO DEVICE TYPE LIST
002060  005176                          .WORD   L$DVTYP
002062                          L$REPP::                ;PTR. TO REPORT CODE
002062  017172                          .WORD   L$RPT
002064                          L$EXP4::
002064  000000                          .WORD   0
002066                          L$EXP5::
002066  000000                          .WORD   0
002070                          L$AUT::                 ;PTR. TO ADD UNIT CODE
002070  000000                          .WORD   0
002072                          L$DUT::                 ;PTR. TO DROP UNIT CODE
002072  000000                          .WORD   0
002074                          L$LUN::                 ;LUN FOR EXERCISERS TO FILL
002074  000000                          .WORD   0
002076                          L$DESP::                ;POINTER TO DIAG. DESCRIPTION
002076  005226                          .WORD   L$DESC
002100                          L$LOAD::                ;GENERATE SPECIAL AUTOLOAD EMT
002100  104035                          EMT     E$LOAD
002102                          L$ETP::                 ;POINTER TO ERRTBL
002102  000000                          .WORD   0
002104                          L$ICP::                 ;PTR. TO INIT CODE
002104  017322                          .WORD   L$INIT
002106                          L$CCP::                 ;PTR. TO CLEAN-UP CODE
002106  020214                          .WORD   L$CLEAN
002110                          L$ACP::                 ;PTR. TO AUTO CODE
002110  020212                          .WORD   L$AUTO
002112                          L$PRT::                 ;PTR. TO PROTECT TABLE
002112  017314                          .WORD   L$PROT
002114                          L$TEST::                ;TEST NUMBER
002114  000000                          .WORD   0
002116                          L$DLY::                 ;DELAY COUNT
002116  000000                          .WORD   0
002120                          L$HIME::                :PTR. TO HIGH MEM
002120  000000                          .WORD   0
```

```
                                .SBTTL  DISPATCH TABLE

                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;      DISPATCH TABLE
                                STARS
                                ;;*****************************************************************

                                        DISPATCH        27
002122  000033                          .WORD   27
002124                          L$DISPATCH::
002124  020240                          .WORD   T1
002126  020556                          .WORD   T2
002130  020702                          .WORD   T3
002132  021430                          .WORD   T4
002134  022216                          .WORD   T5
002136  023006                          .WORD   T6
002140  023326                          .WORD   T7
002142  023676                          .WORD   T8
002144  024052                          .WORD   T9
002146  024520                          .WORD   T10
002150  025040                          .WORD   T11
002152  025320                          .WORD   T12
002154  027534                          .WORD   T13
002156  030234                          .WORD   T14
002160  030750                          .WORD   T15
002162  031270                          .WORD   T16
002164  031550                          .WORD   T17
002166  033544                          .WORD   T18
002170  034250                          .WORD   T19
002172  034770                          .WORD   T20
002174  035336                          .WORD   T21
002176  036606                          .WORD   T22
002200  037014                          .WORD   T23
002202  037222                          .WORD   T24
002204  037510                          .WORD   T25
002206  040236                          .WORD   T26
002210  041006                          .WORD   T27
```

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE

                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;      DEFAULT P-TABLE VALUES
                                STARS
                                ;;*****************************************************************

                                        BGNHW
002212  000004                         .WORD   L10000-L$HW/2
002214                          L$HW::

                                        .WORD   CSRADR          ;DEVICE ADDRESS
                                        .WORD   VECADR          ;INTERRUPT VECTOR ADDRESS
                                        .WORD   BRPRI           ;BUS REQUEST PRIORITY
                                        .WORD   0               ;DRIVE NUMBER

                                        ENDHW
002224                          L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE

002224                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;      SOFTWARE P-TABLE
                                ;;
                                ;;      EACH FLAG CAN BE MODIFIED BY THE OPERATOR
                                ;;      ON START/RESTART COMMAND.  ALL FLAGS ARE
                                ;;      INITIALLY FALSE; A VALUE OF "1" INDICATES
                                ;;      THE CONDITION IS TRUE.
002224                          STARS
                                ;;*****************************************************************

002224                                  BGNSW
002224  000010                          .WORD   L10001-L$SW/2
002226                          L$SW::

002226  000000                  REVFLG::        .WORD   0       ;DR70 REV "A"
002230  000000                  AT0FLG::        .WORD   0       ;DISABLE AT0 ATTN
002232  000000                  AT3FLG::        .WORD   0       ;DISABLE AT3 ATTN
002234  000000                  ABOFLG::        .WORD   0       ;DISABLE AT3 ABORT LOGIC
002236  000000                  BYTFLG::        .WORD   0       ;BYTE MODE TRANSFER
002240  000000                  PARFLG::        .WORD   0       ;GENERATE PARITY ERROR
002242  000000                  DPFLG::         .WORD   0       ;USER DATA PATTERN FLAG
002244  000000                  BWFLG::         .WORD   0       ;ADJUST BAND WIDTH
002246                                  ENDSW
002246                          L10001:

002246                                  ENDMOD

                                .SBTTL
```

```
1444                                    .SBTTL  GLOBAL EQUATES SECTION
1472 002246                                     BGNMOD
1473 002246                             STARS
                                        ;;*****************************************************************
1474                                    ;;
1475                                    ;;      GLOBAL EQUATES
1476 002246                             STARS
                                        ;;*****************************************************************
1477
1478 002246                                     EQUALS
                                        ;
                                        ; BIT DIFINITIONS
                                        ;
                100000                  BIT15== 100000
                040000                  BIT14== 40000
                020000                  BIT13== 20000
                010000                  BIT12== 10000
                004000                  BIT11== 4000
                002000                  BIT10== 2000
                001000                  BIT09== 1000
                000400                  BIT08== 400
                000200                  BIT07== 200
                000100                  BIT06== 100
                000040                  BIT05== 40
                000020                  BIT04== 20
                000010                  BIT03== 10
                000004                  BIT02== 4
                000002                  BIT01== 2
                000001                  BIT00== 1
                                        ;
                001000                  BIT9==  BIT09
                000400                  BIT8==  BIT08
                000200                  BIT7==  BIT07
                000100                  BIT6==  BIT06
                000040                  BIT5==  BIT05
                000020                  BIT4==  BIT04
                000010                  BIT3==  BIT03
                000004                  BIT2==  BIT02
                000002                  BIT1==  BIT01
                000001                  BIT0==  BIT00
                                        ;
                                        ; EVENT FLAG DEFINITIONS
                                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                        ;
                000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
                000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
                000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
                000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
                000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                        ;
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                000340                  PRI07== 340
                000300                  PRI06== 300
                000240                  PRI05== 240
                000200                  PRI04== 200
```

GLOBAL EQUATES SECTION

```
000140                  PRI03== 140
000100                  PRI02== 100
000040                  PRI01== 40
000000                  PRI00== 0
                        ;
                        ;OPERATOR FLAG BITS
                        ;
000004                  EVL==         4
000010                  LOT==        10
000020                  ADR==        20
000040                  IDU==        40
000100                  ISR==       100
000200                  UAM==       200
000400                  BOE==       400
001000                  PNT==      1000
002000                  PRI==      2000
004000                  IXE==      4000
010000                  IBE==     10000
020000                  IER==     20000
040000                  LOE==     40000
100000                  HOE==    100000
```

```
1480 002246                             STARS
                                        ;;*****************************************************************
1481                                    ;;
1482                                    ;;      REGISTER BIT EQUATES:   (REG.***)
1483 002246                             STARS
                                        ;;*****************************************************************
1484
1485                                    ;               CONTROL /STATUS 1 REGISTER BIT EQUATES
1486
1487        000001                      CS1.GO          ==      BIT00           ;GO BIT                 (R/W)   *
1488        000002                      CS1.F0          ==      BIT01           ;FUNCTION BIT 0         (R/W)   *
1489        000004                      CS1.F1          ==      BIT02           ;FUNCTION BIT 1         (R/W)   *
1490        000010                      CS1.F2          ==      BIT03           ;FUNCTION BIT 2         (R/W)   *
1491        000020                      CS1.F3          ==      BIT04           ;FUNCTION BIT 3         (R/W)   *
1492        000040                      CS1.F4          ==      BIT05           ;FUNCTION BIT 4         (R/W)   *
1493        000100                      CS1.IE          ==      BIT06           ;INTERRUPT ENABLE       (R/W)
1494        000200                      CS1.RDY         ==      BIT07           ;READY                  (RO)
1495        000400                      CS1.A16         ==      BIT08           ;UNIBUS ADDR BIT 16     (R/W)
1496        001000                      CS1.A17         ==      BIT09           ;UNIBUS ADDR BIT 17     (R/W)
1497        002000                      CS1.PSEL        ==      BIT10           ;PORT SELECT            (R/W)
1498        004000                      CS1.DVA         ==      BIT11           ;DEVICE AVAILABLE       (RO)    *
1499        020000                      CS1.MCPE        ==      BIT13           ;MB CNTRL BUS PAR ERR   (RO)
1500        040000                      CS1.TRE         ==      BIT14           ;TRANSFER ERROR         (R/W)
1501        100000                      CS1.SC          ==      BIT15           ;SPECIAL CONDITION      (RO)
1502
1503                                    ;               CONTROL /STATUS 2 REGISTER BIT EQUATES
1504
1505        000001                      CS2.U0          ==      BIT00           ;UNIT SELECT BIT 0      (R/W)
1506        000002                      CS2.U1          ==      BIT01           ;UNIT SELECT BIT 1      (R/W)
1507        000004                      CS2.U2          ==      BIT02           ;UNIT SELECT BIT 2      (R/W)
1508        000010                      CS2.BAI         ==      BIT03           ;UB ADDR INCR INHIBIT   (R/W)
1509        000020                      CS2.PAT         ==      BIT04           ;PARITY TEST            (R/W)
1510        000040                      CS2.CLR         ==      BIT05           ;CONTROLER CLEAR        (WO)
1511        000100                      CS2.IR          ==      BIT06           ;INPUT READY            (RO)
1512        000200                      CS2.OR          ==      BIT07           ;OUTPUT READY           (RO)
1513        000400                      CS2.MDPE        ==      BIT08           ;MB DATA BUS PAR ERR    (RO)
1514        001000                      CS2.MXF         ==      BIT09           ;MISSED TRANSFER        (R/W)
1515        002000                      CS2.PGE         ==      BIT10           ;PROGRAM ERROR          (RO)
1516        004000                      CS2.NEM         ==      BIT11           ;NON EXISTENT MEMORY    (RO)
1517        010000                      CS2.NED         ==      BIT12           ;NON EXISTENT DRIVE     (RO)
1518        020000                      CS2.UPE         ==      BIT13           ;UNIBUS PARITY ERROR    (R/W)
1519        040000                      CS2.WCE         ==      BIT14           ;WRITE CHECK ERROR      (RO)
1520        100000                      CS2.DLT         ==      BIT15           ;DATA LATE              (RO)
```

```
1522
1523                            ;               INTERFACE STATUS REGISTER BIT EQUATES
1524
1525        000001              IS.AT0      ==      BIT00       ;USER DEFINED ATTENTION BIT 0
1526        000002              IS.AT1      ==      BIT01       ;USER DEFINED ATTENTION BIT 1
1527        000004              IS.AT2      ==      BIT02       ;USER DEFINED ATTENTION BIT 2
1528        000010              IS.AT3      ==      BIT03       ;USER DEFINED ATTENTION BIT 3
1529        000020              IS.CYC      ==      BIT04       ;PRIME CYCLE FLIP-FLOP BIT
1530        000200              IS.IRY      ==      BIT07       ;INTERFACE READY BIT
1531        000400              IS.IDP      ==      BIT08       ;INTERFACE DATA PARITY ERROR BIT
1532        001000              IS.ICP      ==      BIT09       ;INTERFACE CONTROL PARITY ERROR BIT
1533        002000              IS.ERO      ==      BIT10       ;USER DEFINED ERROR
1534        004000              IS.XABT     ==      BIT11       ;REV "B" TRANSFER ABORT LOGIC
1535        010000              IS.IP       ==      BIT12       ;INVERT PARITY GENERATION
1536        040000              IS.ERR      ==      BIT14       ;ERROR SUM
1537        100000              IS.ATA      ==      BIT15       ;ATTENTION ACTIVE
1538
1539                            ;               FUNCTION / STATUS REGISTER BIT EQUATES
1540
1541        000001              FS.ST0      ==      BIT00       ;STATUS BIT 0           (RO)
1542        000002              FS.ST1      ==      BIT01       ;STATUS BIT 1           (RO)
1543        000004              FS.ST2      ==      BIT02       ;STATUS BIT 2           (RO)
1544        000010              FS.ST3      ==      BIT03       ;STATUS BIT 3           (RO)
1545        000020              FS.ST4      ==      BIT04       ;STATUS BIT 4           (RO)
1546        000040              FS.ST5      ==      BIT05       ;STATUS BIT 5           (RO)
1547        000100              FS.ST6      ==      BIT06       ;STATUS BIT 6           (RO)
1548        000200              FS.ST7      ==      BIT07       ;STATUS BIT 7           (RO)
1549        000400              FS.F0       ==      BIT08       ;FUNCTION BIT 0         (R/W)
1550        001000              FS.F1       ==      BIT09       ;FUNCTION BIT 1         (R/W)
1551        002000              FS.F2       ==      BIT10       ;FUNCTION BIT 2         (R/W)
1552        004000              FS.F3       ==      BIT11       ;FUNCTION BIT 3         (R/W)
1553        010000              FS.F4       ==      BIT12       ;FUNCTION BIT 4         (R/W)
1554        020000              FS.F5       ==      BIT13       ;FUNCTION BIT 5         (R/W)
1555        040000              FS.F6       ==      BIT14       ;FUNCTION BIT 6         (R/W)
1556        100000              FS.F7       ==      BIT15       ;FUNCTION BIT 7         (R/W)
```

```
1558
1559                                    ;               COMMAND DEFINITIONS
1560
1561            000011                  CLR.CMD         ==      11              ;INTERFACE CLEAR COMMAND
1562
1563            000071                  RD.CMD          ==      71              ;READ & INCR ADDR COMMAND
1564            000061                  WR.CMD          ==      61              ;WRITE & INCR ADDR COMMAND
1565            000171                  RIE.CMD         ==      171             ;READ & INCR ADDR WITH INT ENABLED
1566            000161                  WIE.CMD         ==      161             ;WRITE & INCR ADDR WITH INT ENABLED
1567
1568            000077                  RD2.CMD         ==      77              ;READ & DECR ADDR COMMAND
1569            000067                  WR2.CMD         ==      67              ;WRITE & DECR ADDR COMMAND
1570            000177                  RIE2.CMD        ==      177             ;READ & DECR ADDR WITH INT ENABLED
1571            000167                  WIE2.CMD        ==      167             ;WRITE & DECR ADDR WITH INT ENABLED
1572
1573                                    ;               TEST SELECT BIT EQUATES
1574
1575            000200                  B.REVA          ==      BIT07           ;REV "A" SELECT
1576            000020                  B.UPAR          ==      BIT04           ;USER PARITY DISABLE
1577            000010                  B.BYTE          ==      BIT03           ;BYTE MODE XFER SELECT
1578            000004                  B.ABORT         ==      BIT02           ;AT3 ABORT DISABLE
1579            000002                  B.AT3           ==      BIT01           ;AT3 ATTN DISABLE
1580            000001                  B.AT0           ==      BIT00           ;AT0 ATTN DISABLE
1581
1582                                    ;               DEVICE CONFIGURATION
1583
1584            177776                  PS              ==      177776          ;PSW LOCATION
1585            177764                  CPUID           ==      177764          ;SYSTEM ID (PDP-11/70)
1586            000033                  MAXTST          ==      27.             ;CURRENT NUMBER OF TESTS
1587
1588                                    ;               DEFAULT P-TABLE VALUE EQUATES
1589
1590            172440                  CSRADR          ==      172440          ;CSR ADDRESS
1591            000224                  VECADR          ==      224             ;INTERRUPT VECTOR ADDRESS
1592            000005                  BRPRI           ==      5               ;BUS REQUEST PIORITY
```

```
                                        .SBTTL  GLOBAL DATA SECTION
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      REGISTER DISPLACEMENT VALUES:   (D.***)
                                        ;;
                                        ;;          EACH LOCATION CONTAINS THE CORRESPONDING REGISTER'S
                                        ;;          WORD DISPLACEMENT FROM THE DEVICE BASE ADDRESS
                                        STARS
                                        ;;*****************************************************************

                                        D.TBL::                         ;TABLE START

                                        D.CS1::         .WORD   0       ;CONTROL / STATUS 1
                                        D.WC::          .WORD   2       ;WORD COUNT
                                        D.BA::          .WORD   4       ;BUS ADDRESS
                                        D.FS::          .WORD   6       ;FUNCTION / STATUS
                                        D.CS2::         .WORD   10      ;CONTROL / STATUS 2
                                        D.IS::          .WORD   12      ;INTERFACE STATUS
                                        D.IB::          .WORD   14      ;INPUT BUFFER
                                        D.AS::          .WORD   16      ;ATTENTION SUMMARY
                                        D.OB::          .WORD   20      ;OUTPUT BUFFER
                                        D.DT::          .WORD   26      ;DRIVE TYPE
                                        D.BAE::         .WORD   30      ;BUFFER ADDR EXTENSION  *
                                        D.CS3::         .WORD   32      ;CONTROL / STATUS 3     *

                                        D.SIZ           ==      .-D.TBL         ;TABLE SIZE
```

```
1620 002272                            STARS
                                       ;;*****************************************************************
1621                                   ;;
1622                                   ;;      REGISTER ADDRESS STORAGE:       (A.***)
1623 002276                            STARS
                                       ;;*****************************************************************
1624
1625                                   ;               REGISTER ADDRESS STORAGE
1626
1627 002276                            A.TBL::                         ;TABLE START
1628
1629 002276  000000                    A.CS1::         .WORD   0       ;CONTROL / STATUS 1     172440
1630 002300  000000                    A.WC::          .WORD   0       ;WORD COUNT             172442
1631 002302  000000                    A.BA::          .WORD   0       ;BUS ADDRESS            172444
1632 002304  000000                    A.FS::          .WORD   0       ;FUNCTION / STATUS      172446
1633 002306  000000                    A.CS2::         .WORD   0       ;CONTROL / STATUS 2     172450
1634 002310  000000                    A.IS::          .WORD   0       ;INTERFACE STATUS       172452
1635 002312  000000                    A.IB::          .WORD   0       ;INPUT BUFFER           172454
1636 002314  000000                    A.AS::          .WORD   0       ;ATTENTION SUMMARY      172456
1637 002316  000000                    A.OB::          .WORD   0       ;OUTPUT BUFFER          172460
1638 002320  000000                    A.DT::          .WORD   0       ;DRIVE TYPE             172466
1639 002322  000000                    A.BAE::         .WORD   0       ;BUFFER ADDR EXTENSION  172470  *
1640 002324  000000                    A.CS3::         .WORD   0       ;CONTROL / STATUS 3     172472  *
1641
1642         000030                    A.SIZ           ==      .-A.TBL         ;TABLE SIZE
```

```
   1644 002326                          STARS
                                        ;;*****************************************************************
   1645                                 ;;
   1646                                 ;;      REGISTER VALUE STORAGE:         (R.***)
   1647 002326                          STARS
                                        ;;*****************************************************************
   1648
   1649                                 ;               REGISTER VALUE STORAGE
   1650
   1651 002326                          R.TBL::                         ;TABLE START
   1652
   1653 002326  000000                  R.CS1::         .WORD   0
   1654 002330  000000                  R.WC::          .WORD   0
   1655 002332  000000                  R.BA::          .WORD   0
   1656 002334  000000                  R.FS::          .WORD   0
   1657 002336  000000                  R.CS2::         .WORD   0
   1658 002340  000000                  R.IS::          .WORD   0
   1659 002342  000000                  R.IB::          .WORD   0
   1660 002344  000000                  R.AS::          .WORD   0
   1661 002346  000000                  R.OB::          .WORD   0
   1662 002350  000000                  R.DT::          .WORD   0
   1663 002352  000000                  R.BAE::         .WORD   0
   1664 002354  000000                  R.CS3::         .WORD   0
   1665
   1666         000030                  R.SIZ           ==      .-R.TBL         ;TABLE SIZE
```

```
1668 002356                            STARS
                                       ;;*****************************************************************
1669                                   ;;
1670                                   ;;      REGISTER INITIAL VALUE STORAGE:         (IN.***)
1671                                   ;;
1672                                   ;;          EACH LOCATION RESERVES A STORAGE LOCATION CONTAINING
1673                                   ;;          INITIAL REGISTER CONTENTS USED IN THE REGISTER
1674                                   ;;          INITIALIZATION TEST 2.
1675 002356                            STARS
                                       ;;*****************************************************************
1676
1677                                   ;               REGISTER INITIAL CONTENTS STORAGE
1678
1679 002356                            IN.TBL::                                ;TABLE START
1680
1681 002356  004200                    IN.CS1::        .WORD   4200            ;CONTROL / STATUS 1     (DVA & RDY)
1682 002360  000000                    IN.WC::         .WORD   0               ;WORD COUNT
1683 002362  000000                    IN.BA::         .WORD   0               ;BUFFER ADDRESS
1684 002364  000000                    IN.FS::         .WORD   0               ;FUNCTION / STATUS
1685 002366  000100                    IN.CS2::        .WORD   100             ;CONTROL / STATUS 2     (IR)
1686 002370  000200                    IN.IS::         .WORD   200             ;INTERFACE STATUS       (IRY)
1687 002372  000000                    IN.IB::         .WORD   0               ;INPUT BUFFER
1688 002374  000000                    IN.AS::         .WORD   0               ;ATTENTION SUMMARY
1689 002376  000000                    IN.OB::         .WORD   0               ;OUTPUT BUFFER
1690 002400  000000                    IN.DT::         .WORD   0               ;DRIVE TYPE
1691 002402  000000                    IN.BAE::        .WORD   0               ;BUFFER ADDR EXTENSION  *
1692 002404  000000                    IN.CS3::        .WORD   0               ;CONTROL / STATUS 3     *
1693
1694         000030                    IN.SIZ          ==      .-IN.TBL        ;TABLE SIZE
```

```
1696 002406                            STARS
                                       ;;*****************************************************************
1697                                   ;;
1698                                   ;;      REGISTER NAME ADDRESS POINTERS:          (NA.***)
1699                                   ;;
1700                                   ;;          EACH LOCATION CONTAINS A POINTER TO THE REGISTER
1701                                   ;;          NAME FORMAT STRING, USED TO DISPLAY REGISTER NAMES.
1702 002406                            STARS
                                       ;;*****************************************************************
1703
1704                                   ;               REGISTER NAME ADDRESS POINTERS
1705
1706 002406                            NA.TBL::                                ;TABLE START ADDRESS
1707
1708 002406  005352                    NA.CS1::        .WORD   N.CS1           ;POINTER TO ASCII CS1
1709 002410  005362                    NA.WC::         .WORD   N.WC            ;POINTER TO ASCII WC
1710 002412  005372                    NA.BA::         .WORD   N.BA            ;POINTER TO ASCII BA
1711 002414  005402                    NA.FS::         .WORD   N.FS            ;POINTER TO ASCII FS
1712 002416  005412                    NA.CS2::        .WORD   N.CS2           ;POINTER TO ASCII CS2
1713 002420  005422                    NA.IS::         .WORD   N.IS            ;POINTER TO ASCII IS
1714 002422  005432                    NA.IB::         .WORD   N.IB            ;POINTER TO ASCII IB
1715 002424  005442                    NA.AS::         .WORD   N.AS            ;POINTER TO ASCII AS
1716 002426  005452                    NA.OB::         .WORD   N.OB            ;POINTER TO ASCII OB
1717 002430  005462                    NA.DT::         .WORD   N.DT            ;POINTER TO ASCII DT
1718 002432  005472                    NA.BAE::        .WORD   N.BAE           ;POINTER TO ASCII BAE   *
1719 002434  005502                    NA.CS3::        .WORD   N.CS3           ;POINTER TO ASCII CS3   *
1720
1721         000030                    NA.SIZ          ==      .-NA.TBL        ;TABLE SIZE
```

```
 1723 002436                            STARS
                                        ;;*****************************************************************
 1724                                   ;;
 1725                                   ;;      REGISTER BIT NAME ADDRESS POINTERS:             (BA.***)
 1726                                   ;;
 1727                                   ;;          EACH LOCATION CONTAINS A POINTER TO THE REGISTER
 1728                                   ;;          BIT NAME FORMAT STRING, USED BY DSPBITS SUBROUTINE
 1729                                   ;;          TO DISPLAY REGISTER NAMES.
 1730 002436                            STARS
                                        ;;*****************************************************************
 1731
 1732                                   ;               REGISTER BIT NAME ADDRESS POINTERS
 1733
 1734 002436                            BA.TBL::                                ;TABLE START ADDRESS
 1735
 1736 002436  005512                    BA.CS1::        .WORD   B.CS1           ;POINTER TO CS1 REGISTER BIT NAMES
 1737 002440  006134                    BA.WC::         .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1738 002442  006134                    BA.BA::         .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1739 002444  005602                    BA.FS::         .WORD   B.FS            ;POINTER TO FS REGISTER BIT NAMES
 1740 002446  005702                    BA.CS2::        .WORD   B.CS2           ;POINTER TO CS2 REGISTER BIT NAMES
 1741 002450  005776                    BA.IS::         .WORD   B.IS            ;POINTER TO IS REGISTER BIT NAMES
 1742 002452  006134                    BA.IB::         .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1743 002454  006064                    BA.AS::         .WORD   B.AS            ;POINTER TO AS REGISTER BIT NAMES
 1744 002456  006134                    BA.OB::         .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1745 002460  006134                    BA.DT::         .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1746 002462  006134                    BA.BAE::        .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1747 002464  006134                    BA.CS3::        .WORD   B.BIT           ;POINTER TO DEFAULT REGISTER BIT NAMES
 1748
 1749         000030                    BA.SIZ          ==      .-BA.TBL        ;TABLE SIZE
```

```
1751
1752                                    ;               REGISTER TEST SELECTION MASK STORAGE
1753
1754 002466  000000                    R.MAX::         .WORD   0               ;MAXIMUM REGISTER INDEX VALUE
1755 002470  000000                    R.ERR::         .WORD   0
1756
1757 002472  000000                    R.SST::         .WORD   0               ;STATUS REGISTER SELECT
1758 002474  001777                    R.SAD::         .WORD   1777            ;ADDRESS RESPONSE SELECT
1759 002476  001271                    R.SIN::         .WORD   1271            ;INITIALIZTION SELECT
1760 002500  000027                    R.SRH::         .WORD   27              ;CONTROLLER STATUS SELECT
1761 002502  001750                    R.SDR::         .WORD   1750            ;INTERFACE STATUS SELECT
1762
1763                                    ;               DATA PARAMETER STORAGE
1764
1765 002504  000000                    D.FILPTR::      .WORD   0               ;POINTER TO FILE NAME
1766 002506  000000                    D.BLOCK::       .WORD   0               ;TRANSFER BLOCK SIZE
1767 002510  000000                    D.LPCNT::       .WORD   0               ;TRANSFER LOOP COUNTER
1768 002512  000000                    D.MAX::         .WORD   0               ;DATA COMPARISON MAX INDEX
1769 002514  000000                    D.WDP::         .WORD   0               ;USER WORD DATA PATTERN
1770 002516  000000                    D.BDP::         .WORD   0               ;USER BYTE DATA PATTERN
1771
1772                                    ;               REPORT SUMMARY TABLES
1773
1774 002520                            RP.PASS::       .BLKW   32.             ;PASS COUNT TABLE
1775
1776         000100                    RP.SIZ          ==      .-RP.PASS       ;TABLE SIZE
1777
1778 002620                            RP.ERROR::      .BLKW   32.             ;ERROR COUNT TABLE
1779
1780                                    ;               LOCAL P-TABLE VALUES
1781
1782 002720  000000                    PT.CSR::        .WORD   0               ;CSR ADDRESS
1783 002722  000000                    PT.VEC::        .WORD   0               ;VECTOR ADDRESS
1784 002724  000000                    PT.PRI::        .WORD   0               ;BUS REQUEST PRICRITY
1785 002726  000000                    PT.DRI::        .WORD   0               ;DRIVE SELECT
```

```
1787
1788                                    ;               FORMAT PARAMETER STORAGE
1789
1790 002730  000000                    FP.NAM::        .WORD   0               ;REGISTER NAME FORMAT ADDRESS
1791 002732  000000                    FP.ADR::        .WORD   0               ;ADDRESS PARAMETER
1792
1793 002734  000000                    FP.EXP::        .WORD   0               ;EXPECTED VALUE
1794 002736  000000                    FP.ACT::        .WORD   0               ;ACTUAL VALUE
1795 002740  000000                    FP.XOR::        .WORD   0               ;XOR VALUE
1796 002742  000000                    FP.NDX::        .WORD   0               ;DATA INDEX VALUE
1797 002744  000000                    FP.DPAT::       .WORD   0               ;DATA PATTERN
1798
1799 002746  000000                    FP.TBL::        .WORD   0               ;BIT EXPAND TABLE ADDRESS
1800 002750  005054                    FP.BIT::        .WORD   BFR             ;REGISTER BITS PRINT BUFFER ADDRESS
```

```
1802
1803                                    ;               REGISTER SAVE AREA
1804
1805 002752  000000                    REG0::          .WORD   0
1806 002754  000000                    REG1::          .WORD   0
1807 002756  000000                    REG2::          .WORD   0
1808 002760  000000                    REG3::          .WORD   0
1809 002762  000000                    REG4::          .WORD   0
1810 002764  000000                    REG5::          .WORD   0
1811 002766  000000                    REG6::          .WORD   0
1812 002770  000000                    REG7::          .WORD   0
```

```
                                ;               MONITOR LOCATIONS
                                HRDTBL::        .WORD   0               ;P-TABLE ADDRESS
                                LOGUNIT::       .WORD   0               ;LOGICAL UNIT NUMBER
                                UNITPOS::       .WORD   0               ;UNIT BIT POSITION
                                DRIVE::         .WORD   0               ;DRIVE SELECT
                                ERRNDX::        .WORD   0               ;DATA ERROR INDEX
                                ERRCNT::        .WORD   0               ;ERROR COUNTER
                                BITCNT::        .WORD   0               ;BIT EXPAND COUNT

                                SELECT::        .WORD   0               ;OPTION SELECT WORD
                                                                        ;BIT7=  REV "A" BOARD
                                                                        ;BIT4=  INPUT PARITY DISABLE
                                                                        ;BIT3=  BYTE MODE TRANSFER
                                                                        ;BIT2=  AT03 ABORT DISABLE
                                                                        ;BIT1=  AT03 ATTN DISABLE
                                                                        ;BIT0=  AT00 ATTN DISABLE

                                ;               FLAG WORDS
                                CPUFLG::        .WORD   0               ;PDP-11/70 CPU FLAG
                                INIFLG::        .WORD   0               ;DIAGNOSTIC INITIALIZE FLAG
                                HLPFLG::        .WORD   0               ;HELP REQUEST PROMPT FLAG
                                TABFLG::        .WORD   0               ;TABULAR COUNT
                                INTFLG::        .WORD   0               ;INTERRUPT FLAG
                                TRPFLG::        .WORD   0               ;ADDRESS TRAP FLAG
                                ERRFLG::        .WORD   0               ;ERROR FLAG / COUNTER
                                MBCFLG::        .WORD   0               ;MASSBUS CONTROLLER ERROR FLAG
                                DRIFLG::        .WORD   0               ;INTERFACE ERROR FLAG

                                ;               DELAY SUBROUTINE PARAMETERS

                                DLYCNT::        .WORD   0       ;SPARE DELAY COUNTER
                                DMACNT::        .WORD   10      ;DMA DELAY COUNT:        T = 96 USEC
                                SILOCNT::       .WORD   12      ;SILO FILL DELAY COUNT: T = 16-128 USEC

                                ;               MESSAGE PRINT FLAGS

                                CSRFLG::        .WORD   0               ;CSR ADDRESS MSG FLAG
                                NEDFLG::        .WORD   0               ;DRIVE SELECT MSG FLAG
                                PWRFLG::        .WORD   0               ;POWER ON MSG FLAG
                                CBLFLG::        .WORD   0               ;CABLE MSG FLAG
                                PERFLG::        .WORD   0               ;PARITY MSG FLAG

                                ;       BUFFERS

                                IBUF::          .BLKW   400             ;INPUT BUFFER   (256 WORDS)
                                OBUF::          .BLKW   400             ;OUTPUT BUFFER  (256 WORDS)

                001000          BFRSIZ  ==      . - OBUF                ;BUFFER SIZE

                                BFR::           .BLKW   44              ;TEXT BUFFER    (72 BYTES)
                                                .WORD   0               ;EOB
```

```
                                .SBTTL  ERROR TABLE DEFINITION

                                        ERRTBL
005166                          L$ERRTBL::
005166  000000                  ERRTYP::        .WORD   0
005170  000000                  ERRNBR::        .WORD   0
005172  000000                  ERRMSG::        .WORD   0
005174  000000                  ERRBLK::        .WORD   0
```

GLOBAL TEXT SECTION

```
1873                                    .SBTTL  GLOBAL TEXT SECTION
1874 005176                             STARS
                                        ;;*****************************************************************
1875                                    ;;
1876                                    ;;      GLOBAL TEXT SECTION
1877 005176                             STARS
                                        ;;*****************************************************************
1878
1879                                            .NLIST  BEX                             ;DON'T EXPAND STRINGS
1880
1881                                    ;       NAMES OF DEVICES SUPPORTED BY PROGRAM
1882
1883 005176                                     DEVTYP  <DR70 MASSBUS INTERFACE>
     005176                             L$DVTYP::
     005176  104  122  067                      .ASCIZ  /DR70 MASSBUS INTERFACE/
                                                .EVEN
1884
1885                                    ;       TEST DESCRIPTION
1886
1887 005226                                     DESCRIPT        <DR70 REPAIR DIAGNOSTIC>
     005226                             L$DESC::
     005226  104  122  067                      .ASCIZ  /DR70 REPAIR DIAGNOSTIC/
                                                .EVEN
1888
1889                                    ;               HELP FILE PRINT BUFFER
1890
1891 005256  045  101                  LINE::          .ASCII  /%A/            ;ALPHA NUMERIC BUFFER FLAG
1892 005260                             LINBUF::        .BLKB   70              ;SEVENTY CHARACTER BUFFER
1893 005350  000000                     EOL::           .WORD   0               ;END OF LINE
```

```
1895 005352                            STARS
                                       ;;*****************************************************************
1896                                   ;;
1897                                   ;;      REGISTER NAME FORMAT STRINGS:          (N.***)
1898                                   ;;
1899 005352                            STARS
                                       ;;*****************************************************************
1900
1901                                   ;               REGISTER NAMES
1902
1903 005352                            N.TBL::                                        ;TABLE START ADDRESS
1904
1905 005352    122    110    056      N.CS1::         .ASCIZ          /RH.CS1/
1906                                                  .EVEN
1907 005362    122    110    056      N.WC::          .ASCIZ          /RH.WC /
1908                                                  .EVEN
1909 005372    122    110    056      N.BA::          .ASCIZ          /RH.BA /
1910                                                  .EVEN
1911 005402    104    122    056      N.FS::          .ASCIZ          /DR.FS /
1912                                                  .EVEN
1913 005412    122    110    056      N.CS2::         .ASCIZ          /RH.CS2/
1914                                                  .EVEN
1915 005422    104    122    056      N.IS::          .ASCIZ          /DR.IS /
1916                                                  .EVEN
1917 005432    104    122    056      N.IB::          .ASCIZ          /DR.IB /
1918                                                  .EVEN
1919 005442    104    122    056      N.AS::          .ASCIZ          /DR.AS /
1920                                                  .EVEN
1921 005452    104    122    056      N.OB::          .ASCIZ          /DR.OB /
1922                                                  .EVEN
1923 005462    104    122    056      N.DT::          .ASCIZ          /DR.DT /
1924                                                  .EVEN
1925 005472    122    110    056      N.BAE::         .ASCIZ          /RH.BAE/
1926                                                  .EVEN
1927 005502    122    110    056      N.CS3::         .ASCIZ          /RH.CS3/
1928                                                  .EVEN
1929
1930           000140                  N.SIZ           ==      .-N.TBL                ;TABLE SIZE
```

```
1932 005512                            STARS
                                       ;;*****************************************************************
1933                                   ;;
1934                                   ;;      REGISTER BIT NAME DEFINITIONS          (B.***)
1935 005512                            STARS
                                       ;;*****************************************************************
1936
1937 005512      123     103     054   B.CS1:: .ASCIZ  /SC,TRE,MCPE,,DVA,PSEL,A17,A16,RDY,IE,F4,F3,F2,F1,F0,GO/
1938                                           .EVEN
1939 0056U6      106     122     067   B.FS::  .ASCII  /FR7,FR6,FR5,FR4,FR3,FR2,FR1,FR0,/
1940 005642      104     123     067           .ASCIZ  /DS7,DS6,DS5,DS4,DS3,DS2,DS1,DS0/
1941                                           .EVEN
1942 005702      104     114     124   B.CS2:: .ASCIZ  /DLT,WCE,UPE,NED,NEM,PGE,MXF,MDPE,OR,IR,CLR,PAT,BAI,U2,U1,U0/
1943                                           .EVEN
1944 005776      101     124     101   B.IS::  .ASCIZ  /ATA,ERR,,,XABT,ERO,ICP,IDP,IRY,,,CYC,AT3,AT2,AT1,AT0/
1945                                           .EVEN
1946 006064      054     054     054   B.AS::  .ASCIZ  /........,AS7,AS6,AS5,AS4,AS3,AS2,AS1,AS0/
1947                                           .EVEN
1948 006134      102     061     065   B.BIT:: .ASCII  /B15,B14,B13,B12,B11,B10,B09,B08,/
1949 006174      102     060     067           .ASCIZ  /B07,B06,B05,B04,B03,B02,B01,B00/
1950                                           .EVEN
```

```
1952
1953                                    ;               ERROR MESSAGES
1954
1955 006234    122     105     107  M.ADR1::          .ASCIZ  'REGISTER ADDRESS ERROR'
1956 006263    116     117     116  M.ADR2::          .ASCIZ  'NON-EXISTENT DRIVE ERROR'
1957
1958 006314    101     124     124  M.ASB1::          .ASCIZ  'ATTENTION SUMMARY BIT NOT SET BY UNIT UNDER TEST'
1959
1960 006375    116     117     040  M.ABT1::          .ASCIZ  'NO WORD TRANSFER ABORT ON INTERFACE WRITE'
1961 006447    116     117     040  M.ABT2::          .ASCIZ  'NO WORD TRANSFER ABORT ON INTERFACE READ'
1962 006520    116     117     040  M.ABT3::          .ASCIZ  'NO BYTE TRANSFER ABORT ON INTERFACE WRITE'
1963 006572    116     117     040  M.ABT4::          .ASCIZ  'NO BYTE TRANSFER ABORT ON INTERFACE READ'
1964
1965 006643    101     124     101  M.ATA1::          .ASCIZ  'ATA NOT SET ON USER ATTN CONDITION'
1966 006706    101     124     101  M.ATA2::          .ASCIZ  'ATA NOT SET BY ERR ACTIVE'
1967 006740    101     124     101  M.ATA3::          .ASCIZ  'ATA NOT RESET BY INTERFACE CLEAR CMD'
1968 007005    116     117     040  M.ATA4::          .ASCIZ  'NO INTERRUPT WHEN INTERFACE STATUS ATA SET'
1969 007060    101     124     101  M.ATA5::          .ASCIZ  'ATA NOT RESET BY WRITING TO ATTN SUMMARY BIT'
1970 007135    101     124     101  M.ATA6::          .ASCIZ  'ATA NOT RESET BY LOADING DATA TRANSFER CMD'
1971
1972 007210    103     123     061  M.CSR1::          .ASCIZ  'CS1 FUNCTION BIT ERROR'
1973 007237    103     123     061  M.CSR2::          .ASCIZ  'CS1 REGISTER STATUS ERROR'
1974
1975 007271    127     117     122  M.WRD1::          .ASCIZ  'WORD TRANSFER NOT COMPLETE'
1976 007324    127     117     122  M.WRD2::          .ASCIZ  'WORD TRANSFER ERROR ON INTERFACE WRITE'
1977 007373    127     117     122  M.WRD3::          .ASCIZ  'WORD TRANSFER ERROR ON INTERFACE READ'
1978 007441    127     117     122  M.WRD4::          .ASCIZ  'WORD TRANSFER COMPARISON ERROR'
1979
1980 007500    102     131     124  M.BYT1::          .ASCIZ  'BYTE TRANSFER NOT COMPLETE'
1981 007533    102     131     124  M.BYT2::          .ASCIZ  'BYTE TRANSFER ERROR ON INTERFACE WRITE'
1982 007602    102     131     124  M.BYT3::          .ASCIZ  'BYTE TRANSFER ERROR ON INTERFACE READ'
1983 007650    102     131     124  M.BYT4::          .ASCIZ  'BYTE TRANSFER COMPARISON ERROR'
1984
1985 007707    104     101     124  M.DLT1::          .ASCIZ  'DATA LATE ERROR; ADJUST BANDWIDTH'
1986
1987 007751    105     122     117  M.ERO1::          .ASCIZ  'ERO NOT SET THROUGH M8432 TEST MODULE'
1988 010017    105     122     117  M.ERO2::          .ASCIZ  'ERO NOT RESET BY INTERFACE CLEAR CMD'
1989
1990 010064    105     122     122  M.ERR1::          .ASCIZ  'ERR NOT SET BY ERO ACTIVE'
1991 010116    105     122     122  M.ERR2::          .ASCIZ  'ERR NOT RESET BY INTERFACE CLEAR CMD'
1992
1993 010163    106     125     116  M.FSR1::          .ASCIZ  'FUNCTION STATUS BIT ERROR'
1994
1995 010215    111     116     124  M.IDP1::          .ASCIZ  'INTERFACE STATUS IDP NOT SET BY DATA PARITY ERROR'
1996 010277    111     116     124  M.IDP2::          .ASCIZ  'INTERFACE STATUS IDP NOT RESET BY INTERFACE CLEAR CMD'
1997 010365    111     116     124  M.IDP3::          .ASCIZ  'INTERFACE STATUS IDP NOT RESET BY WRITE TO IDP BIT'
1998
1999 010450    111     116     124  M.ICP1::          .ASCIZ  'INTERFACE STATUS ICP NOT SET BY CONTROL PARITY ERROR'
2000 010535    111     116     124  M.ICP2::          .ASCIZ  'INTERFACE STATUS ICP NOT RESET BY INTERFACE CLEAR CMD'
2001 010623    111     116     124  M.ICP3::          .ASCIZ  'INTERFACE STATUS ICP NOT RESET BY WRITE TO ICP BIT'
2002
2003 010706    116     117     040  M.INT1::          .ASCIZ  'NO INTERRUPT ON WORD TRANSFER END OF BLOCK'
2004 010761    116     117     040  M.INT2::          .ASCIZ  'NO INTERRUPT ON BYTE TRANSFER END OF BLOCK'
```

```
2006
2007 011034    111    116    124  M.IRY1::        .ASCIZ  "INTERFACE STATUS IRY NOT SET ON ERROR"
2008 011102    111    116    124  M.IRY2::        .ASCIZ  "INTERFACE STATUS IRY BIT RESET WHEN CYC BIT SET"
2009 011162    111    116    124  M.IRY3::        .ASCIZ  "INTERFACE STATUS IRY NOT RESET BY DATA TRANSFER CMD"
2010 011246    111    122    131  M.IRY4::        .ASCIZ  "IRY NOT SET BY WORD TRANSFER END OF BLOCK"
2011 011320    111    122    131  M.IRY5::        .ASCIZ  "IRY NOT SET BY WORD TRANSFER ABORT"
2012 011363    111    122    131  M.IRY6::        .ASCIZ  "IRY NOT SET BY INTERFACE CLEAR CMD"
2013 011426    111    122    131  M.IRY7::        .ASCIZ  "IRY NOT SET BY BYTE TRANSFER END OF BLOCK"
2014 011500    111    122    131  M.IRY8::        .ASCIZ  "IRY NOT SET BY BYTE TRANSFER ABORT"
2015
2016 011543    111    116    124  M.ISR1::        .ASCIZ  "INTERFACE STATUS ERROR"
2017 011572    111    116    124  M.ISR2::        .ASCIZ  "INTERFACE STATUS ERROR ON INTERFACE CLEAR CMD"
2018
2019 011650    103    117    116  M.MBC1::        .ASCIZ  "CONTROLLER NOT CLEARED BY MASSBUS INIT"
2020 011717    115    101    123  M.MBC2::        .ASCIZ  "MASSBUS CONTROLLER ERROR"
2021
2022 011750    115    101    123  M.RDY1::        .ASCIZ  "MASSBUS CONTROLLER RDY BIT NOT RESET BY IRY"
2023 012024    115    101    123  M.RDY2::        .ASCIZ  "MASSBUS CONTROLLER RDY BIT NOT SET BY IRY"
2024
2025 012076    122    105    107  M.REG1::        .ASCIZ  "REGISTER INITIALIZATION ERROR"
2026 012134    122    105    107  M.REG2::        .ASCIZ  "REGISTER ERROR"
2027
2028 012153    123    127    064  M.SW4::         .ASCIZ  "SW4 DOES NOT DISABLE AT0 INPUT TO INTERFACE ATA"
2029 012233    123    127    065  M.SW5::         .ASCIZ  "SW5 DOES NOT DISABLE AT3 INPUT TO INTERFACE ATA"
2030 012313    123    127    066  M.SW6::         .ASCIZ  "SW6 DOES NOT DISABLE AT3 TRANSFER ABORT LOGIC"
2031
2032 012371    103    117    116  M.TRE1::        .ASCIZ  "CONTROLLER TRE BIT NOT SET BY TRANSFER ABORT LOGIC"
2033
2034 012454    111    116    124  M.UAT1::        .ASCIZ  "INTERFACE STATUS USER ATTN BIT NOT SET/RESET"
2035 012531    111    116    124  M.UAT2::        .ASCIZ  "INTERFACE STATUS USER ATTENTION BIT NOT RESET"
```

```
2037 012607                            STARS
                                       ;;*****************************************************************
2038                                   ;;
2039                                   ;;      ERROR FORMAT STRINGS:                    (F.***)
2040 012607                            STARS
                                       ;;*****************************************************************
2041
2042                                   ;               FORMAT STATEMENTS
2043
2044 012607    045    124    045  F.NAM::        .ASCIZ  '%T%S2'
2045                                              .EVEN
2046 012616    045    117    066  F.VAL::        .ASCIZ  '%O6%S2'
2047                                              .EVEN
2048 012626    045    116    061  F.ADR1::       .ASCIZ  '%N1%AREG%S5%AADRS'
2049                                              .EVEN
2050 012650    045    116    061  F.ADR2::       .ASCIZ  '%N1%T%S2%O6'
2051                                              .EVEN
2052 012664    045    116    061  F.DMP1::       .ASCIZ  '%N1%ACONTROLLER STATUS%N1'
2053                                              .EVEN
2054 012716    045    116    061  F.DMP2::       .ASCIZ  '%N1%AINTERFACE STATUS%N1'
2055                                              .EVEN
2056 012750    045    116    061  F.DMP3::       .ASCIZ  '%N1%O6%S2%O6%S2%O6%S2%O6%S2%O6%S2%O6%N1'
2057                                              .EVEN
2058 013020    045    116    061  F.DMP4::       .ASCIZ  '%N1%T%S2%A[%O6%A]%S2%O6%S2%A;'
2059                                              .EVEN
2060 013056    045    116    061  F.REG1::       .ASCIZ  '%N1%AREG%S5%AACT%S5%AEXP%S5%AXOR%S5%ABIT(S)'
2061                                              .EVEN
2062 013132    045    116    061  F.REG2::       .ASCIZ  '%N1%T%S2%O6%S2%O6%S2%O6%S2'
2063                                              .EVEN
2064 013166    045    116    061  F.DAT1::       .ASCIZ  '%N1%AADRS%S4%AWRD%S3%ARECVD%S3%AEXPTD%S3%AXOR'
2065                                              .EVEN
2066 013244    045    116    061  F.DAT2::       .ASCIZ  '%N1%O6%S2%D3%S3%O6%S2%O6%S2%O6'
2067                                              .EVEN
2068 013304    045    116    062  F.ERCNT::      .ASCIZ  '%N2%ATOTAL ERRORS:  %D3%N1'
2069                                              .EVEN
2070 013340    045    116    061  F.MODE::       .ASCIZ  /%N1%ACHECK SW1-7 SET TO 'OFF' POSITION%N1/
2071                                              .EVEN
2072 013412    045    116    061  F.PERR::       .ASCIZ  /%N1%ACHECK SW1-8 SET TO 'ON' POSITION%N1/
2073                                              .EVEN
2074 013464    045    116    061  F.CBL::        .ASCIZ  '%N1%ACHECK M8432 TEST MODULE LOOP-BACK CABLE%N1'
2075                                              .EVEN
2076 013544    045    116    061  F.BW::         .ASCIZ  '%N1%ABANDWIDTH ADJUST:  TYPE CNTRL/C TO EXIT%N1'
2077                                              .EVEN
2078 013624    045    116    061  F.DPAT::       .ASCIZ  '%N1%ADATA PATTERN:  %O6%N1'
2079                                              .EVEN
2080 013660    045    116    061  F.CSR::        .ASCIZ  '%N1%ACHECK MASSBUS CONTROLLER CSR ADDRESS'
2081                                              .EVEN
2082 013732    045    116    061  F.NED::        .ASCIZ  '%N1%ACHECK DRIVE SELECT SWITCHES 1-3'
2083                                              .EVEN
2084 014000    045    116    061  F.PWR::        .ASCIZ  '%N1%ACHECK UNIT FOR POWER ON'
2085                                              .EVEN
2086 014036    045    116    061  F.UNIT::       .ASCIZ  '%N1%AP-TABLE ERROR ON UNIT%D4'
2087                                              .EVEN
```

```
2089
2090 014074  045  116  061  F.RPT1::   .ASCII  '%N1%ADIAGNOSTIC SUMMARY:'
2091 014124  045  116  062             .ASCIZ  '%N2%ATEST%S4%APASSES%S2%AERRORS'
2092                                    .EVEN
2093 014164  045  116  061  F.RPT2::   .ASCIZ  '%N1%D2%S6%D6%S2%D6'
2094                                    .EVEN
2095 014210  045  124  000  F.BITS::   .ASCIZ  '%T'
2096                                    .EVEN
2097 014214  045  116  000  F.CRLF::   .ASCIZ  '%N'
2098                                    .EVEN
2099
2100 014220  105  116  124  Q.WDP::    .ASCIZ  'ENTER TEST DATA WORD'
2101                                    .EVEN
2102 014246  105  116  124  Q.BDP::    .ASCIZ  'ENTER TEST DATA BYTE'
2103                                    .EVEN
2104 014274  127  117  122  Q.WBL::    .ASCIZ  'WORDS PER BLOCK'
2105                                    .EVEN
2106 014314  102  131  124  Q.BBL::    .ASCIZ  'BYTES PER BLOCK'
2107                                    .EVEN
2108
2109                        .          SUPERVISOR HELP FILE
2110 014334  120  122  111  Q.HLP1::   .ASCII  'PRINT SUPERVISOR HELP FILE:  '
2111 014371  104  122  123  D.FIL1::   .ASCIZ  'DRS.HLP'
2112                                    .EVEN
2113
2114                        .          DIAGNOSTIC HELP FILE
2115 014402  120  122  111  Q.HLP2::   .ASCII  'PRINT DIAGNOSTIC HELP FILE:  '
2116 014437  132  104  122  D.FIL2::   .ASCIZ  'ZDRMA0.HLP'
2117                                    .EVEN
2118
2119                                    .LIST   BEX                    ;EXPAND
2120                                    .ENABL  AMA
```

```
                                .SBTTL  DEVICE ADDRESS ERROR ROUTINE
014452                          ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;;
                                ;;  REPORT ADDRESS ERROR        (EXTENDED PRINT)
                                ;;
                                ;;      INPUT:  A.TBL::         REGISTER ADDRESS POINTER TABLE
                                ;;              R.ERR::         REGISTER ADDRESS ERROR MASK
                                ;;              R.MAX::         MAXIMUM REGISTER TABLE INDEX
                                ;;
                                ;;      OUTPUT: ERRCNT::        INCREMENT
014452                          ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                        BGNMSG  ER.ADR
014452                          ER.ADR::
014452  010237  002756                MOV     R2,REG2         ;SAVE REGISTER 2
014456  010437  002762                MOV     R4,REG4         ;SAVE REGISTER 4

014462  005002                        CLR     R2              ;CLEAR REGISTER INDEX
014464  012704  000001                MOV     #1,R4           ;INITIALIZE REGISTER TEST BIT

014470                                  PRINTB  #F.ADR1         ;PRINT ADDRESS ERROR HEADER
014470  012746  012626                MOV     #F.ADR1,-(SP)
014474  012746  000001                MOV     #1,-(SP)
014500  010600                        MOV     SP,R0
014502  104414                        TRAP    C$PNTB
014504  062706  000004                ADD     #4,SP

014510  030457  002470          100$:   BIT     R4,R.ERR        ;TEST ERROR MASK
014514  001424                        BEQ     200$            ;BRANCH IF BIT NOT SET

014516  005237  003004                INC     ERRCNT                  ;BUMP ERROR COUNT
014522  016237  002406  002730        MOV     NA.TBL(R2),FP.NAM       ;LOAD REGISTER NAME ADDRESS
014530  016237  002276  002732        MOV     A.TBL(R2),FP.ADR        ;SAVE REGISTER ADDRESS

014536                                  PRINTB  #F.ADR2,FP.NAM,FP.ADR   ;PRINT ADDRESS ERROR
014536  013746  002732                MOV     FP.ADR,-(SP)
014542  013746  002730                MOV     FP.NAM,-(SP)
014546  012746  012650                MOV     #F.ADR2,-(SP)
014552  012746  000003                MOV     #3,-(SP)
014556  010600                        MOV     SP,R0
014560  104414                        TRAP    C$PNTB
014562  062706  000010                ADD     #10,SP

014566  006304          200$:   ASL     R4              ;SHIFT TEST BIT TO NEXT REGISTER
014570  062702  000002                ADD     #2,R2           ;BUMP INDEX COUNTER
014574  020237  002466                CMP     R2,R.MAX        ;AND COMPARE WITH MAX INDEX VALUE
014600  002743                        BLT     100$            ;CONTINUE IF LESS THAN MAXIMUM

014602                                  PRINTB  #F.CRLF ;PRINT CARRAGE RETURN/LINE FEED
014602  012746  014214                MOV     #F.CRLF,-(SP)
014606  012746  000001                MOV     #1,-(SP)
014612  010600                        MOV     SP,R0
014614  104414                        TRAP    C$PNTB
014616  062706  000004                ADD     #4,SP
```

```
   2159
   2160 014622  013702  002756                  MOV     REG2,R2         ;RESTORE REGISTER 2
   2161 014626  013704  002762                  MOV     REG4,R4         ;RESTORE REGISTER 4
   2162
   2163 014632                                  EXIT    MSG
        014632  000167                          .WORD   J$JMP
        014634  000000                          .WORD   L10002-2-.
   2164
   2165 014636                                  ENDMSG
        014636                          L10002:
        014636  104423                          TRAP    C$MSG
```

REGISTER INITIALIZATION ERROR ROUTINE

```
                                        .SBTTL  REGISTER INITIALIZATION ERROR ROUTINE
014640                                  STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  REPORT REGISTER INITIALIZATION ERROR        (EXTENDED PRINT)
                                        ;;
                                        ;;      INPUT:  A.TBL::         REGISTER ADDRESS POINTER TABLE
                                        ;;              R.ERR::         REGISTER ADDRESS ERROR MASK
                                        ;;
                                        ;;      OUTPUT: ERRCNT::        INCREMENT
                                        ;;
                                        ;;      EXTERNALS:              DSPREG
014640                                  STARS
                                        ;;*****************************************************************

                                                BGNMSG  ER.INI
014640                                  ER.INI::

014640  010237  002756                          MOV     R2,REG2         ;SAVE REGISTER 2
014644  010437  002762                          MOV     R4,REG4         ;SAVE REGISTER 4

014650  005002                                  CLR     R2              ;CLEAR REGISTER INDEX
014652  012704  000001                          MOV     #1,R4           ;INITIALIZE REGISTER TEST BIT

014656                                          PRINTX  #F.REG1                 ;PRINT REGISTER ERROR HEADER
014656  012746  013056                          MOV     #F.REG1,-(SP)
014662  012746  000001                          MOV     #1,-(SP)
014666  010600                                  MOV     SP,R0
014670  104415                                  TRAP    C$PNTX
014672  062706  000004                          ADD     #4,SP

014676  030437  002470          100$:           BIT     R4,R.ERR        ;TEST ERROR MASK
014702  001420                                  BEQ     200$            ;BRANCH IF BIT NOT SET

014704  005237  003004                          INC     ERRCNT                  ;BUMP ERROR COUNT
014710  016237  002406  002730                  MOV     NA.TBL(R2),FP.NAM       ;MOVE REGISTER NAME ADDRESS
014716  016237  002326  002736                  MOV     R.TBL(R2),FP.ACT        ;MOVE ACTUAL VALUE
014724  016237  002356  002734                  MOV     IN.TBL(R2),FP.EXP       ;MOVE EXPECTED VALUE
014732  016237  002436  002746                  MOV     BA.TBL(R2),FP.TBL       ;PARAMETERS

014740  004737  016054                          JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS

014744  006304                  200$:           ASL     R4              ;SHIFT TEST BIT TO NEXT REGISTER
014746  062702  000002                          ADD     #2,R2           ;BUMP INDEX COUNTER
014752  020237  002466                          CMP     R2,R.MAX        ;AND COMPARE WITH MAX INDEX VALUE
014756  002747                                  BLT     100$            ;CONTINUE IF LESS
```

REGISTER INITIALIZATION ERROR ROUTINE

```
2206
2207 014760                                    PRINTX  #F.CRLF ;PRINT CARRAGE RETURN/LINE FEED
     014760  012746  014214                    MOV     #F.CRLF,-(SP)
     014764  012746  000001                    MOV     #1,-(SP)
     014770  010600                            MOV     SP,R0
     014772  104415                            TRAP    C$PNTX
     014774  062706  000004                    ADD     #4,SP
2208
2209 015000  013702  002756                    MOV     REG2,R2         ;RESTORE REGISTER 2
2210 015004  013704  002762                    MOV     REG4,R4         ;RESTORE REGISTER 4
2211
2212 015010                                    EXIT    MSG
     015010  000167                            .WORD   J$JMP
     015012  000000                            .WORD   L10003-2-.
2213
2214 015014                                    ENDMSG
     015014                            L10003:
     015014  104423                            TRAP    C$MSG
```

```
                                .SBTTL  REGISTER VALUE ERROR ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  REPORT REGISTER VALUE ERROR         (EXTENDED PROINT)
                                ;;
                                ;;      INPUT:  FP.NAM::        POINTER TO NAME FORMAT
                                ;;              FP.EXP::        EXPECTED REGISTER CONTENTS
                                ;;              FP.ACT::        ACTUAL REGISTER CONTENTS
                                ;;              FP.TBL::        BIT EXPAND TABLE ADDRESS
                                ;;              FP.BIT::        PRINT BUFFER ADDRESS
                                ;;
                                ;;      OUTPUT: ERRCNT::        INCREMENT
                                ;;
                                ;;      EXTERNALS:              DSPREG
                                STARS
                                ;;*****************************************************************

                                        BGNMSG  ER.REG
                                ER.REG::

                                        PRINTX  #F.REG1                 ;PRINT REGISTER ERROR HEADER
015016  012746  013056                  MOV     #F.REG1,-(SP)
015022  012746  000001                  MOV     #1,-(SP)
015026  010600                          MOV     SP,R0
015030  104415                          TRAP    C$PNTX
015032  062706  000004                  ADD     #4,SP

015036  005237  003004                  INC     ERRCNT                  ;BUMP ERROR COUNT
015042  004737  016054                  JSR     PC,DSPREG               ;DISPALY REGISTER CONTENTS

015046                                  PRINTX  #F.CRLF                 ;NEW LINE
015046  012746  014214                  MOV     #F.CRLF,-(SP)
015052  012746  000001                  MOV     #1,-(SP)
015056  010600                          MOV     SP,R0
015060  104415                          TRAP    C$PNTX
015062  062706  000004                  ADD     #4,SP

015066                                  EXIT    MSG                     ;AND RETURN
015066  000167                          .WORD   J$JMP
015070  000000                          .WORD   L10004-2-.

015072                                  ENDMSG
015072                          L10004:
015072  104423                          TRAP    C$MSG
```

REGISTER DUMP ROUTINE

```
                                        .SBTTL  REGISTER DUMP ROUTINE
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  REGISTER DUMP ROUTINE       (EXTENDED PRINT)
                                        ;;
                                        ;;      INPUT:  R.SRH::         CONTROLLER REGISTER SELECT MASK
                                        ;;              R.SDR::         INTERFACE REGISTER SELECT MASK
                                        ;;
                                        ;;      OUTPUT: ERRCNT::        INCREMENT
                                        ;;
                                        ;;      EXTERNALS:              DSPSTAT
                                        STARS
                                        ;;*****************************************************************

                                                BGNMSG  ER.DMP
                                        ER.DMP::
015074                          100$:   PRINTB  #F.DMP1                 ;PRINT CONTROLLER HEADER
015074  012746  012664                  MOV     #F.DMP1,-(SP)
015100  012746  000001                  MOV     #1,-(SP)
015104  010600                          MOV     SP,R0
015106  104414                          TRAP    C$PNTB
015110  062706  000004                  ADD     #4,SP

015114  005237  003004                  INC     ERRCNT                  ;BUMP ERROR COUNT
015120  013737  002500  002472          MOV     R.SRH,R.SST             ;LOAD CONTROLLER REGISTER SELECT MASK
015126  004737  016164                  JSR     PC,DSPSTAT              ;DISPLAY CONTROLLER REGISTERS

015132                                  PRINTB  #F.DMP2                 ;PRINT INTERFACE HEADER
015132  012746  012716                  MOV     #F.DMP2,-(SP)
015136  012746  000001                  MOV     #1,-(SP)
015142  010600                          MOV     SP,R0
015144  104414                          TRAP    C$PNTB
015146  062706  000004                  ADD     #4,SP

015152  013737  002502  002472          MOV     R.SDR,R.SST             ;LOAD INTERFACE REGISTER SELECT MASK
015160  004737  016164                  JSR     PC,DSPSTAT              ;DISPLAY INTERFACE REGISTERS

015164                                  EXIT    MSG                     ;AND RETURN
015164  000167                          .WORD   J$JMP
015166  000000                          .WORD   L10005-2-.

015170                                  ENDMSG
015170                          L10005:
015170  104423                          TRAP    C$MSG
```

```
                                .SBTTL  DATA TRANSFER ERROR ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;      REPORT DATA TRANSFER ERROR      (EXTENDED PRINT)
                                ;;
                                ;;      INPUT:          NONE
                                ;;
                                ;;      OUTPUT:         PRINTS HEADER ONLY
                                STARS
                                ;;*****************************************************************

                                        BGNMSG  ER.DATA
                                ER.DATA::
                                        PRINTX  #F.DAT1                 ;PRINT DATA ERROR HEADER
015172  012746  013166                  MOV     #F.DAT1,-(SP)
015176  012746  000001                  MOV     #1,-(SP)
015202  010600                          MOV     SP,R0
015204  104415                          TRAP    C$PNTX
015206  062706  000004                  ADD     #4,SP

015212                                  EXIT    MSG                     ;AND RETURN
015212  000167                          .WORD   J$JMP
015214  000000                          .WORD   L10006-2-.

                                        ENDMSG
                                L10006:
015216  104423                          TRAP    C$MSG
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION
015220                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;      ADDRESS TIME-OUT SERVICE ROUTINE
                                ;;
                                ;;      OUTPUT: TRPFLG::        INCREMENT
015220                          STARS
                                ;;*****************************************************************

                                        BGNSRV  TRAP4
015220                          TRAP4::
015220  005237  003024                  INC     TRPFLG          ;SET TRAP FLAG

                                        ENDSRV                  ;RETURN FROM INTERRUPT
015224                          L10007:
015224  000002                          RTI
015226                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;      DEVICE INTERRUPT SERVICE ROUTINE
                                ;;
                                ;;      OUTPUT: INTFLG::        INCREMENT
015226                          STARS
                                ;;*****************************************************************

                                        BGNSRV  INTSRV
015226                          INTSRV::
015226  005237  003022                  INC     INTFLG          ;SET INTERRUPT FLAG

                                        ENDSRV                  ;RETURN FROM INTERRUPT
015232                          L10010:
015232  000002                          RTI
```

```
                                  .SBTTL  SET/CLEAR  TRAP VECTOR
     015234                       STARS
                                  ;;*************************************************************
                                  ;;
                                  ;;  DESCRIPTION:        SET TRAP VECTOR
                                  ;;
                                  ;;  CALLING SEQUENCE:   JSR     PC,SETRAP
     015234                       STARS
                                  ;;*************************************************************

     015234                       SETRAP::

     015234                               SETVEC  #4,#TRAP4,#PRI07          ;SETUP VECTOR
     015234  012746  000340               MOV     #PRI07,-(SP)
     015240  012746  015220               MOV     #TRAP4,-(SP)
     015244  012746  000004               MOV     #4,-(SP)
     015250  012746  000003               MOV     #3,-(SP)
     015254  104437                       TRAP    C$SVEC
     015256  062706  000010               ADD     #10,SP

     015262  000207                       RTS     PC                        ;RETURN
     015264                       STARS
                                  ;;*************************************************************
                                  ;;
                                  ;;  DESCRIPTION:        CLEAR TRAP VECTOR
                                  ;;
                                  ;;  CALLING SEQUENCE:   JSR     PC,CLRTRAP
     015264                       STARS
                                  ;;*************************************************************

     015264                       CLRTRAP::

     015264                               CLRVEC  #4                        ;CLEAR VECTOR
     015264  012700  000004               MOV     #4,R0
     015270  104436                       TRAP    C$CVEC

     015272  000207                       RTS     PC                        ;RETURN
```

```
                                       .SBTTL  SET/CLEAR  INTERRUPT SERVICE ROUTINE VECTOR
 015274                                STARS
                                       ;;*****************************************************************
                                       ;;
                                       ;;  DESCRIPTION:        SET INTERRUPT SERVICE ROUTINE VECTOR
                                       ;;
                                       ;;  CALLING SEQUENCE:   JSR     PC,SETISR
 015274                                STARS
                                       ;;*****************************************************************

 015274                                SETISR::

 015274                                        SETVEC  PT.VEC,#INTSRV,PT.PRI    ;SETUP VECTOR
 015274  013746  002724                        MOV     PT.PRI,-(SP)
 015300  012746  015226                        MOV     #INTSRV,-(SP)
 015304  013746  002722                        MOV     PT.VEC,-(SP)
 015310  012746  000003                        MOV     #3,-(SP)
 015314  104437                                TRAP    C$SVEC
 015316  062706  000010                        ADD     #10,SP

 015322  000207                                RTS     PC                       ;RETURN
 015324                                STARS
                                       ;;*****************************************************************
                                       ;;
                                       ;;  DESCRIPTION:        CLEAR INTERRUPT SERVICE ROUTINE VECTOR
                                       ;;
                                       ;;  CALLING SEQUENCE:   JSR     PC,CLRISR
 015324                                STARS
                                       ;;*****************************************************************

 015324                                CLRISR::

 015324  112777  000000  164744                MOVB    #0,@A.CS1                ;CLEAR MBC INTERRUPT ENABLE

 015332                                        CLRVEC  PT.VEC                   ;CLEAR VECTOR
 015332  013700  002722                        MOV     PT.VEC,R0
 015336  104436                                TRAP    C$CVEC

 015340  000207                                RTS     PC                       ;RETURN
```

```
                                        .SBTTL  MASSBUS CONTROLLER CLEAR
015342                                  STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  DESCRIPTION:         CLEAR MASSBUS CONTROLLER AND TEST STATUS
                                        ;;
                                        ;;      STEPS:           SET CONTROLLER CLEAR BIT IN CS2 REGISTER
                                        ;;                       RESET DRIVE SELECT IN CS2 REGISTER
                                        ;;                       READ REGISTERS AND COMPARE CS1 REGISTER WITH EXP VALUE
                                        ;;
                                        ;;      OUTPUT:          MBCFLG::  INCREMENT IF UNEXPECTED STATUS (ERROR)
                                        ;;
                                        ;;      CALLING SEQ:     JSR      PC,CLRMBC
                                        ;;
                                        ;;      EXTERNALS:       DLY,RDREG
015342                                  STARS
                                        ;;*****************************************************************

015342                                  CLRMBC::
015342  005037  003030                          CLR     MBCFLG                  ;RESET ERROR FLAG
015346  012777  000040  164732                  MOV     #CS2.CLR,@A.CS2         ;SET CONTROLLER CLEAR
015354  013777  003000  164724                  MOV     DRIVE,@A.CS2            ;RESET UNIT SELECT

015362  004737  016042                          JSR     PC,DLY                  ;DELAY A FEW CYCLES
015366  004737  015606                          JSR     PC,RDREG                ;READ REGISTERS

015372  022737  004200  002326                  CMP     #4200,R.CS1             ;COMPARE EXPECTED STATUS WITH NEW
015400  001402                                  BEQ     100$                    ;BRANCH IF IDENTICAL

015402  005237  003030                          INC     MBCFLG                  ;UNEXPECTED STATUS:  INCR ERROR FLAG

015406  000207                          100$:   RTS     PC                      ;AND RETURN
```

```
                                   .SBTTL  TEST MASSBUS CONTROLLER STATUS
     015410                        STARS
                                   ;;************************************************************
                                   ;;
                                   ;;  DESCRIPTION:          TEST MASSBUS CONTROLLER FOR READY AND NO ERRORS
                                   ;;
                                   ;;      STEPS:            CLEAR MBC ERROR FLAG AND READ REGISTERS
                                   ;;                        TEST FOR CS1 REGISTER RDY BIT SET
                                   ;;                        TEST FOR CS1 REGISTER SC STATUS BIT RESET
                                   ;;
                                   ;;      INPUT:            R.CS1:  CONTENTS OF CS1 REGISTER
                                   ;;
                                   ;;      OUTPUT:           INCREMENT MBCFLG IF SC STATUS SET (ERROR)
                                   ;;
                                   ;;      CALLING SEQ:      JSR     PC,TSTMBC
                                   ;;
                                   ;;      EXTERNALS:        RDREG
     015410                        STARS
                                   ;;************************************************************

     015410                        TSTMBC::
     015410  005037  003030                CLR     MBCFLG                  ;CLEAR ERROR FLAG
     015414  004737  015606                JSR     PC,RDREG                ;READ REGISTERS

     015420  013700  002326                MOV     R.CS1,R0                ;RELOAD CS1 REGISTER
     015424  042700  000076                BIC     #76,R0                  ;CLEAR FUNCTION BITS F4->F0
     015430  022700  004200                CMP     #4200,R0                ;EXPECTED:  DVA & RDY
     015434  001402                        BEQ     100$                    ;BRANCH IF IDENTICAL

     015436  005237  003030                INC     MBCFLG                  ;UNEXPECTED STATUS:  INCR ERROR FLAG

     015442  000207                100$:   RTS     PC                      ;AND RETURN
```

```
2436                                    .SBTTL  TEST INTERRUPT RECEIVED
2437 015444                             STARS
                                        ;;*****************************************************************
2438                                    ;;
2439                                    ;;  DESCRIPTION:        TIME OUT ON TEST FOR INTERRUPT RECEIVED
2440                                    ;;
2441                                    ;;      STEPS:          LOAD TIME OUT VALUE
2442                                    ;;                      REPEAT UNTIL TIME OUT:
2443                                    ;;                        IF INTERRUPT RECEIVED:  EXIT
2444                                    ;;
2445                                    ;;      INPUT:          INTFLG::        SET TO ZERO
2446                                    ;;
2447                                    ;;      OUTPUT:         NONE
2448                                    ;;
2449                                    ;;      CALLING SEQ:    JSR     PC,TSTINT
2450                                    ;;
2451                                    ;;      EXTERNALS:      NONE
2452 015444                             STARS
                                        ;;*****************************************************************
2453
2454 015444                             TSTINT::
2455
2456 015444  012700  002000                     MOV     #2000,R0                ;LOAD TIME-OUT VALUE
2457
2458 015450  005737  003022             100$:   TST     INTFLG                  ;TEST INTERRUPT RECEIVED
2459 015454  001002                             BNE     200$                    ;EXIT IF NOT ZERO
2460
2461 015456  005300                             DEC     R0                      ;DECREMENT COUNTER
2462 015460  001373                             BNE     100$                    ;AND CONTINUE IF NOT ZERO
2463
2464 015462  000207                     200$:   RTS     PC                      ;AND RETURN
```

```
                                        .SBTTL  DR70 INTERFACE CLEAR
                                        ;+
                                        ;*****************************************************************
                                        ;
                                        ;  DESCRIPTION:        CLEAR DR70 INTERFACE AND TEST IRY STATUS
                                        ;
                                        ;       STEPS:          ISSUE INTERFACE CLEAR COMMAND THROUGH CS1
                                        ;                       READ REGISTERS AND TEST INTERFACE STATUS
                                        ;
                                        ;       OUTPUT:         INCREMENT CLRDRI FLAG IF IRY NOT SET (ERROR)
                                        ;
                                        ;       CALLING SEQ:    JSR     PC,CLRDRI
                                        ;
                                        ;       EXTERNALS:      DLY,RDREG
                                        ;-
                                        ;*****************************************************************

                                        CLRDRI::
015464  005037  003032                          CLR     DRIFLG                  ;CLEAR ERROR FLAG
015470  012777  000011  164600                  MOV     #CLR.CMD,@A.CS1         ;ISSUE INTERFACE CLEAR COMMAND
015476  013777  003000  164602                  MOV     DRIVE,@A.CS2            ;RESET UNIT SELECT
015504  012777  000020  164576                  MOV     #IS.CYC,@A.IS           ;AND SET CYCLE REQUEST
015512  004737  016042                          JSR     PC,DLY                  ;DELAY A FEW MICROSECONDS
015516  004737  015606                          JSR     PC,RDREG                ;READ REGISTERS

015522  032737  000001  002326                  BIT     #CS1.GO,R.CS1           ;TEST GO BIT RESET (REV "A")
015530  001004                                  BNE     50$                     ;ERROR IF GO NOT RESET

015532  022737  000200  002340                  CMP     #IS.IRY,R.IS            ;AND COMPARE WITH IRY
015540  001402                                  BEQ     100$                    ;BRANCH IF INTERFACE STATUS GOOD

015542  005237  003032                50$:      INC     DRIFLG                  ;ERROR:  SET FLAG

015546  000207                        100$:     RTS     PC                      ;AND RETURN
```

```
2499                                    .SBTTL  CLEAR TEST ROUTINE
2500 015550                             STARS
                                        ;;**********************************************************************
2501                                    ;;
2502                                    ;;      CLEAR TEST ROUTINE
2503                                    ;;
2504                                    ;;      INPUT   L$TEST::        CURRENT TEST NUMBER
2505                                    ;;              ERRCNT::        TEST ERROR COUNT
2506                                    ;;
2507                                    ;;      OUTPUT  RP.PASS::       INCREMENT
2508                                    ;;              P.ERRCR::       SUMMED
2509                                    ;;              ERRCNT::        CLEARED
2510                                    ;;
2511                                    ;;      CALLING SEQ:            JSR     PC,CLRTST
2512 015550                             STARS
                                        ;;**********************************************************************
2513
2514 015550                             CLRTST::
2515
2516                                    ;       COMPUTE TEST INDEX
2517 015550  013700  002114                    MOV     L$TEST,R0               ;LOAD TEST NO.
2518 015554  005300                            DEC     R0                      ;SUBTRACT ONE
2519 015556  006300                            ASL     R0                      ;TIMES TWO FOR WORD INDEX
2520
2521 015560  022700  000100                    CMP     #RP.SIZ,R0              ;QC TEST INDEX
2522 015564  003407                            BLE     200$                    ;EXIT IF TOO LARGE
2523
2524                                    ;       UPDATE REPORT TABLES
2525 015566  005260  002520            100$:   INC     RP.PASS(R0)             ;INCREMENT PASS COUNT
2526 015572  063760  003004  002620            ADD     ERRCNT,RP.ERROR(R0)     ;SUM TEST ERROR COUNT
2527 015600  005037  003004                    CLR     ERRCNT                  ;CLEAR ERROR COUNT
2528
2529 015604  000207                    200$:   RTS     PC                      ;RETURN
```

```
                                        .SBTTL  READ REGISTER CONTENTS ROUTINE
015606                                  STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  DESCRIPTION:        ROUTINE TO READ REGISTER CONTENTS
                                        ;;
                                        ;;      INPUT:  R.MAX::         MAXIMUM REGISTER INDEX
                                        ;;              A.TBL::         REGISTER ADDRESS TABLE
                                        ;;
                                        ;;      OUTPUT: R.TBL::         TABLE OF REGISTER VALUES
                                        ;;
                                        ;;  CALLING SEQUENCE:   JSR     PC,RDREG
015606                                  STARS
                                        ;;*****************************************************************

015606                                  RDREG::
015606  010237  002756                          MOV     R2,REG2                 ;SAVE REGISTER 2
015612  005002                                  CLR     R2                      ;CLEAR INDEX REGISTER

015614  017262  002276  002326  100$:    MOV     @A.TBL(R2),R.TBL(R2)    ;READ REGISTER
015622  062702  000002                          ADD     #2,R2                   ;BUMP INDEX
015626  020237  002466                          CMP     R2,R.MAX                ;AND COMPARE WITH MAX INDEX
015632  002770                                  BLT     100$                    ;BRANCH IF LESS

015634  013702  002756                          MOV     REG2,R2                 ;RESTORE REGISTER
015640  000207                                  RTS     PC                      ;RETURN
```

```
                                .SBTTL  BLOCK READ ROUTINE
                                STARS
                                ;;*************************************************************
                                ;;
                                ;;  DESCRIPTION:        ROUTINE TO READ A 64 WORD BLOCK FROM THE INTERFACE
                                ;;
                                ;;      STEPS:          INITIALIZE FOR READ
                                ;;                      ISSUE READ W/INT ENABLED AND SET CYCLE REQUEST
                                ;;                      TEST FOR INTERRUPT RECEIVED OR TIME OUT
                                ;;
                                ;;      OUTPUT:         NONE
                                ;;
                                ;;      EXTERNALS:      RDINI,TSTINT
                                ;;
                                ;;      CALLING SEQ:    JSR     PC,RDBLK
                                STARS
                                ;;*************************************************************

                                RDBLK::

                                        CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
                                        JSR     PC,RDINI                ;INITIALIZE FOR READ

                                100$:   MOV     #RIE.CMD,@A.CS1         ;ISSUE INTERFACE READ
                                        MOV     #IS.CYC,@A.IS           ;AND SET CYCLE REQUEST

                                        JSR     PC,TSTINT               ;TEST INTERRUPT OR TIME OUT

                                        RTS     PC                      ;RETURN
```

```
                                .SBTTL  BLOCK WRITE ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DESCRIPTION:        ROUTINE TO WRITE ONE 64 WORD BLOCK FROM THE INTERFACE
                                ;;
                                ;;      STEPS:          INITIALIZE FOR WRITE
                                ;;                      ISSUE WRITE W/INT ENABLED AND SET CYCLE REQUEST
                                ;;                      TEST FOR INTERRUPT RECEIVED OR TIME OUT
                                ;;
                                ;;      OUTPUT:         NONE
                                ;;
                                ;;      EXTERNALS:      WRTINI,TSTINT
                                ;;
                                ;;      CALLING SEQ:    JSR     PC,WRTBLK
                                STARS
                                ;;*****************************************************************

                                WRTBLK::

015674  005037  003022                CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
015700  004737  015762                JSR     PC,WRTINI               ;INITIALIZE FOR WRITE

015704  012777  000161  164364  100$:  MOV     #WIE.CMD,@A.CS1         ;ISSUE INTERFACE WRITE
015712  012777  000020  164370        MOV     #IS.CYC,@A.IS           ;AND SET CYCLE REQUEST

015720  004737  015444                JSR     PC,TSTINT               ;TEST INTERRUPT OR TIME OUT

015724  000207                        RTS     PC                      ;RETURN
```

```
                                        .SBTTL  DR70 INTERFACE READ INITIALIZE
 015726                                 STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  DESCRIPTION:          ROUTINE TO INITIALIZE READ FROM DR70 INTERFACE
                                        ;;
                                        ;;      STEPS:            CLEAR INTERRUPT FLAG
                                        ;;                        LOAD WORD COUNT REGISTER
                                        ;;                        LOAD INPUT BUFFER ADDRESS INTO BUS ADDRESS REGISTER
                                        ;;                        ZERO BUS ADDRESS EXTENTION REGISTER FOR 11/70 CPU'S
                                        ;;
                                        ;;      INPUT:            D.BLOCK::        TRANSFER BLOCK SIZE
                                        ;;
                                        ;;      OUTPUT:           NONE
 015726                                 STARS
                                        ;;*****************************************************************

 015726                                 RDINI::
 015726  005037  003022                         CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
 015732  010077  164342                         MOV     R0,@A.WC                ;INITIALIZE NUMBER WORDS TO READ
 015736  012777  003054  164336                 MOV     #IBUF,@A.BA             ;LOAD INPUT BUFFER ADDRESS

 015744  005737  003012                         TST     CPUFLG                  ;TEST FOR 11/70 CPU
 015750  001403                                 BEQ     100$                    ;NO:  CONTINUE

 015752  012777  000000  164342                 MOV     #0,@A.BAE               ;11/70:  ZERO BUFFER ADDRESS EXTENTION

 015760  000207                         100$:   RTS     PC                      ;RETURN
```

```
                                        .SBTTL  DR70 INTERFACE WRITE INITIALIZE
015762                                  STARS
                                        ;;*************************************************************
                                        ;;
                                        ;;  DESCRIPTION:          ROUTINE TO INITIALIZE WRITE FROM DR70 INTERFACE
                                        ;;
                                        ;;      STEPS:            CLEAR INTERRUPT FLAG
                                        ;;                        LOAD WORD COUNT REGISTER
                                        ;;                        LOAD OUTPUT BUFFER ADDRESS INTO BUS ADDRESS REGISTER
                                        ;;                        ZERO BUS ADDRESS EXTENTION REGISTER FOR 11/70 CPU'S
                                        ;;
                                        ;;      INPUT:            D.BLOCK::        TRANSFER BLOCK SIZE
                                        ;;
                                        ;;      OUTPUT:           NONE
015762                                  STARS
                                        ;;*************************************************************

015762                                  WRTINI::
015762  005037  003022                          CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
015766  010077  164306                          MOV     R0,@A.WC                ;INITIALIZE NUMBER WORDS TO READ
015772  012777  004054  164302                  MOV     #OBUF,@A.BA             ;LOAD OUTPUT BUFFER ADDRESS

016000  005737  003012                          TST     CPUFLG                  ;TEST FOR 11/70 CPU
016004  001403                                  BEQ     100$                    ;NO:  CONTINUE

016006  012777  000000  164306                  MOV     #0,@A.BAE               ;11/70:  ZERO BUFFER ADDRESS EXTENTION

016014  000207                          100$:   RTS     PC                      ;RETURN
```

```
                                .SBTTL  DELAY ROUTINES
016016                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DESCRIPTION:        DMA TRANSFER DELAY ROUTINE
                                ;;
                                ;;  CALLING SEQUENCE:   JSR     PC,DMADLY
                                ;;
                                ;;  INPUT:              DMACNT=DELAY COUNT FOR THIS CPU.
                                ;;  OUTPUT:             NONE
                                ;;  REGISTER USAGE:     R0=SCRATCH
016016                          STARS
                                ;;*****************************************************************

016016                          DMADLY::

016016  013700  003036                  MOV     DMACNT,R0       ;INITIALIZE DELAY COUNT
016022                          10$:
016022  005300                          DEC     R0              ;DECREMENT COUNT REGISTER
016024  001376                          BNE     10$             ;CONTINUE UNTIL ZERO

016026  000207                          RTS     PC              ;RETURN

016030                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DESCRIPTION:        SILO FILL DELAY ROUTINE
                                ;;
                                ;;  CALLING SEQUENCE:   JSR     PC,SILODLY
                                ;;
                                ;;  INPUT:              SILOCNT=DELAY COUNT FOR THIS RH ADAPTER.
                                ;;  OUTPUT:             NONE
                                ;;  REGISTER USAGE:     R0=SCRATCH
016030                          STARS
                                ;;*****************************************************************

016030                          SILODLY::

016030  013700  003040                  MOV     SILOCNT,R0      ;INITIALIZE DELAY COUNT
016034                          10$:
016034  005300                          DEC     R0              ;DECREMENT COUNT REGISTER
016036  001376                          BNE     10$             ;CONTINUE UNTIL ZERO

016040  000207                          RTS     PC              ;RETURN

016042                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DESCRIPTION:        ROUTINE TO DELAY A FEW CYCLES
                                ;;
                                ;;  CALLING SEQUENCE:   JSR     PC,DLY
                                ;;
                                ;;  INTPUT:             NONE
                                ;;  OUTPUT:             NONE
                                ;;  REGISTER USAGE:     R0=SCRATCH
016042                          STARS
                                ;;*****************************************************************
```

```
   2717
   2718 016042                          DLY::
   2719 016042  012700  000004                  MOV     #4,R0           ;LOAD DELAY COUNT
   2720 016046                          10$:
   2721 016046  005300                          DEC     R0              ;DECREMENT COUNT
   2722 016050  001376                          BNE     10$             ;CONTINUE UNTIL ZERO
   2723
   2724 016052  000207                          RTS     PC              ;RETURN
```

```
                                .SBTTL  REGISTER DISPLAY ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DISPLAY REGISTER CONTENTS           (EXTENDED PRINT)
                                ;;
                                ;;      INPUT:          FP.NAM::        REGISTER NAME PTR
                                ;;                      FP.ACT::        ACTUAL VALUE
                                ;;                      FP.EXP::        EXPECTED VALUE
                                ;;                      FP.BFR::        BIT EXPAND TABLE PTR
                                ;;
                                ;;      OUTPUT:         FP.XOR::        COMPUTED XOR VALUE
                                ;;
                                ;;      EXTERNALS:      DSPBITS
                                STARS
                                ;;*****************************************************************

                                DSPREG::

016054  013700  002734                  MOV     FP.EXP,R0               ;LOAD EXPECTED VALUE
016060  013737  002736  002740          MOV     FP.ACT,FP.XOR           ;LOAD ACTUAL VALUE
016066  074037  002740                  XOR     R0,FP.XOR               ;COMPUTE XOR VALUE

016072                                  PRINTB  #F.REG2,FP.NAM,FP.ACT,FP.EXP,FP.XOR     ;PRINT VALUES
016072  013746  002740                  MOV     FP.XOR,-(SP)
016076  013746  002734                  MOV     FP.EXP,-(SP)
016102  013746  002736                  MOV     FP.ACT,-(SP)
016106  013746  002730                  MOV     FP.NAM,-(SP)
016112  012746  013132                  MOV     #F.REG2,-(SP)
016116  012746  000005                  MOV     #5,-(SP)
016122  010600                          MOV     SP,R0
016124  104414                          TRAP    C$PNTB
016126  062706  000014                  ADD     #14,SP

016132  004737  017006                  JSR     PC,DSPBITS              ;EXPAND BITS

016136                                  PRINTB  #F.BITS,FP.BIT          ;PRINT BITS
016136  013746  002750                  MOV     FP.BIT,-(SP)
016142  012746  014210                  MOV     #F.BITS,-(SP)
016146  012746  000002                  MOV     #2,-(SP)
016152  010600                          MOV     SP,R0
016154  104414                          TRAP    C$PNTB
016156  062706  000006                  ADD     #6,SP

016162  000207                          RTS     PC                      ;RETURN
```

```
                                .SBTTL  REGISTER STATUS DISPLAY ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DISPLAY REGISTER STATUS                 (EXTENDED PRINT)
                                ;;
                                ;;      INPUT:          R.SST::         REGISTER SELECT MASK
                                ;;                      R.MAX::         MAX REGISTER TABLE INDEX
                                ;;                      R.TBL::         REGISTER VALUE TABLE
                                ;;                      NA.TBL::        REGISTER NAME TABLE
                                ;;                      A.TBL::         REGISTER ADDRESS TABLE
                                ;;                      BA.TBL::        BIT DEFINITION TABLE
                                ;;
                                ;;      CALLING SEQ:    JSR     PC,DSPSTAT
                                STARS
                                ;;*****************************************************************

                                DSPSTAT::

                                        MOV     R2,REG2                 ;SAVE
                                        MOV     R4,REG4                 ;REGISTERS

                                ;       PRINT REGISTER NAMES

                                100$:   CLR     R2                      ;CLEAR TABLE INDEX
                                        MOV     #1,R4                   ;INITIALIZE SELECT BIT

                                125$:   BIT     R4,R.SST                ;TEST SELECT MASK
                                        BEQ     150$                    ;BRANCH IF BIT RESET

                                        PRINTB  #F.DMP4,NA.TBL(R2),A.TBL(R2),R.TBL(R2)  ;PRINT NAME,ADDR,VALUE
016210  016246  002326                  MOV     R.TBL(R2),-(SP)
016214  016246  002276                  MOV     A.TBL(R2),-(SP)
016220  016246  002406                  MOV     NA.TBL(R2),-(SP)
016224  012746  013020                  MOV     #F.DMP4,-(SP)
016230  012746  000004                  MOV     #4,-(SP)
016234  010600                          MOV     SP,R0
016236  104414                          TRAP    C$PNTB
016240  062706  000012                  ADD     #12,SP

016244  005762  002326                  TST     R.TBL(R2)               ;TEST VALUE
016250  001426                          BEQ     150$                    ;SKIP BITS IF ZERO

016252  022762  006134  002436          CMP     #B.BIT,BA.TBL(R2)       ;TEST FOR NULL BIT DEFS
016260  001422                          BEQ     150$                    ;IF NULL SKIP DISPLAY

016262  016237  002326  002740          MOV     R.TBL(R2),FP.XOR        ;LOAD VALUE
016270  016237  002436  002746          MOV     BA.TBL(R2),FP.TBL       ;LOAD TABLE ADDRESS
016276  004737  017006                  JSR     PC,DSPBITS              ;FORMAT BITS
                                        PRINTB  #F.BITS,FP.BIT          ;PRINT BITS
016302  013746  002750                  MOV     FP.BIT,-(SP)
016306  012746  014210                  MOV     #F.BITS,-(SP)
016312  012746  000002                  MOV     #2,-(SP)
016316  010600                          MOV     SP,R0
016320  104414                          TRAP    C$PNTB
016322  062706  000006                  ADD     #6,SP
```

```
 2796 016326  006304                150$:   ASL     R4                      ;SHIFT SELECT
 2797 016330  062702  000002                ADD     #2,R2                   ;INCREMENT INDEX
 2798 016334  020237  002466                CMP     R2,R.MAX                ;COMPARE WITH MAX
 2799 016340  002720                        BLT     125$                    ;CONTINUE IF LESS THAN MAX
 2800
 2801 016342                                PRINTB  #F.CRLF                 ;NEW LINE
      016342  012746  014214                MOV     #F.CRLF,-(SP)
      016346  012746  000001                MOV     #1,-(SP)
      016352  010600                        MOV     SP,R0
      016354  104414                        TRAP    C$PNTB
      016356  062706  000004                ADD     #4,SP
 2802
 2803 016362  013702  002756          300$:  MOV     REG2,R2                 ;RESTORE
 2804 016366  013704  002762                MOV     REG4,R4                 ;REGISTERS
 2805
 2806 016372  000207                        RTS     PC                      ;RETURN
```

```
                                .SBTTL  DATA TRANSFER ERROR ROUTINE
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  DISPLAY DATA TRANSFER ERROR          (EXTENDED PRINT)
                                ;;
                                ;;      INPUT:  FP.NDX::                 BUFFER INDEX
                                ;;              FP.EXP::                 EXPECTED (SENT) DATA VALUE
                                ;;              FP.ACT::                 ACTUAL (RECEIVED) DATA VALUE
                                ;;
                                ;;      OUTPUT: FP.ADR::                 DATA ADDRESS
                                ;;              FP.NDX::                 WORD NUMBER
                                ;;              FP.XOR::                 EXCLUSIVE OR VALUE
                                ;;
                                ;;      CALLING SEQ:            JSR      PC,DSPDATA
                                STARS
                                ;;*****************************************************************

016374                          DSPDATA::
016374  013700  002734                  MOV     FP.EXP,R0               ;LOAD EXPECTED VALUE
016400  013737  002736  002740          MOV     FP.ACT,FP.XOR           ;LOAD ACTUAL VALUE
016406  074037  002740                  XOR     R0,FP.XOR               ;COMPUTE EXCLUSIVE OR

016412  012737  003054  002732          MOV     #IBUF,FP.ADR            ;LOAD BUFFER START ADDRESS
016420  063737  002742  002732          ADD     FP.NDX,FP.ADR           ;ADD INDEX

016426  013700  002742                  MOV     FP.NDX,R0               ;LOAD INDEX
016432  006200                          ASR     R0                      ;DIVIDE BY TWO
016434  005200                          INC     R0                      ;RELATIVE TO ONE
016436  010037  002742                  MOV     R0,FP.NDX               ;SAVE AS WORD NUMBER

016442                          PRINTX  #F.DAT2,FP.ADR,FP.NDX,FP.ACT,FP.EXP,FP.XOR
016442  013746  002740          MOV     FP.XOR,-(SP)
016446  013746  002734          MOV     FP.EXP,-(SP)
016452  013746  002736          MOV     FP.ACT,-(SP)
016456  013746  002742          MOV     FP.NDX,-(SP)
016462  013746  002732          MOV     FP.ADR,-(SP)
016466  012746  013244          MOV     #F.DAT2,-(SP)
016472  012746  000006          MOV     #6,-(SP)
016476  010600                  MOV     SP,R0
016500  104415                  TRAP    C$PNTX
016502  062706  000016          ADD     #16,SP

016506  000207                  RTS     PC
```

```
                                        .SBTTL  DATA TRANSFER ERROR ROUTINE
     016510                             STARS
                                        ;;*****************************************************************

                                        ;;
                                        ;;  DISPLAY DATA TRANSFER ERROR         (EXTENDED PRINT)
                                        ;;
                                        ;;      INPUT:          FP.DPAT::       DATA PATTERN
                                        ;;                      F.DPAT::        DISPLAY FORMAT ADDRESS
                                        ;;
                                        ;;      OUTPUT:         DATA PATTERN DISPLAYED
                                        ;;
                                        ;;      CALLING SEQ:    JSR     PC,DSPDPAT
     016510                             STARS
                                        ;;*****************************************************************

     016510                             DSPDPAT::

     016510                                     PRINTB  #F.DPAT,FP.DPAT         ;DISPLAY DATA
     016510  013746  002744                     MOV     FP.DPAT,-(SP)
     016514  012746  013624                     MOV     #F.DPAT,-(SP)
     016520  012746  000002                     MOV     #2,-(SP)
     016524  010600                             MOV     SP,R0
     016526  104414                             TRAP    C$PNTB
     016530  062706  000006                     ADD     #6,SP

     016534  000207                             RTS     PC
```

HELP FILE DUMP ROUTINE

```
                                .SBTTL  HELP FILE DUMP ROUTINE
016536                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  SUBROUTINE TO DUMP DOS HELP FILE TO CONSOLE
                                ;;
                                ;;      INPUT:          D.FILPTR::      POINTER TO 'FILNAM.EXT''
                                ;;
                                ;;      OUTPUT:         PRINTS 72 CHARACTER BUFFER TO CONSOLE
                                ;;
                                ;;      REGISTER USAGE: R1  =   POINTER TO LINE TEXT BUFFER
                                ;;                      R2  =   TAB POSITION COUNTER
                                ;;                      R4  =   CHARACTER FROM FILE
                                ;;
                                ;;      CALLING SEQ:    MOV     #FILNAM,D.FILPTR
                                ;;                      JSR     PC,DPSFILE
                                ;;
                                ;;      FUNCTION:       DPSFILE WILL OPEN, READ AND CLOSE THE SELETED FILE
                                ;;                      USING THE DIAG SUPERVISOR MACROS.  CHARACTERS
                                ;;                      ARE READ ONE AT A TIME AND TRANSFERRED TO THE
                                ;;                      LINE BUFFER, WHICH IS DUMPED TO THE CONSOLE
                                ;;                      WHEN FULL.
016536                          STARS
                                ;;*****************************************************************

000040          BLK     =       40                      ;BLANK CHAR
000015          CR      =       15                      ;CARRAGE RETURN CHAR
000012          LF      =       12                      ;LINE FEED CHAR
000011          TAB     =       11                      ;TAB CHAR
```

```
2888 016536                            DSPFILE::
2889
2890 016536  010137  002754                    MOV     R1,REG1                 ;SAVE
2891 016542  010237  002756                    MOV     R2,REG2                 ;REGISTERS
2892 016546  010437  002762                    MOV     R4,REG4
2893 016552  005037  003020                    CLR     TABFLG                  ;CLEAR TAB ACTIVE FLAG
2894
2895                                   ;        OPEN FILE AND PRINT NULL LINE
2896
2897 016556                            100$:   OPEN    D.FILPTR                ;OPEN FILE
     016556  013700  002504                    MOV     D.FILPTR,R0
     016562  104434                            TRAP    C$OPEN
2898 016564  012701  005260                    MOV     #LINBUF,R1              ;AND LOAD POINTER TO BUFFER
2899 016570  112721  000015                    MOVB    #CR,(R1)+               ;MOVE CARRAGE RETURN
2900 016574  112721  000012                    MOVB    #LF,(R1)+               ;AND LINE FEED TO BUFFER
2901 016600  112704  000000                    MOVB    #0,R4                   ;FOLLOWED BY NULL
2902 016604  004737  016714                    JSR     PC,600$                 ;PRINT BUFFER, RESET R1
2903
2904                                   ;        GET NEXT CHARACTER FROM SUPERVISOR, AND TEST FOR END OF FILE
2905
2906 016610                            200$:   GETBYTE R4                      ;GET NEXT CHAR IN R4
     016610  104426                            TRAP    C$GETB
     016612  110004                            MOVB    R0,R4
2907 016614                                    BNCOMPLETE 400$                 ;EOF IF INCOMPLETE
     016614  103013                            BCC     400$
2908 016616  001414                            BEQ     425$                    ;OR IF CHARACTER = ZERO
2909
2910                                   ;        TEST AND PROCESS TAB CHARACTER
2911
2912 016620  122704  000011            300$:   CMPB    #TAB,R4                 ;COMPARE WITH TAB
2913 016624  001004                            BNE     350$                    ;NO:  CONTINUE
2914 016626  112704  000040                    MOVB    #BLK,R4                 ;YES:  LOAD R4 WITH A BLANK
2915 016632  005237  003020                    INC     TABFLG                  ;AND SET TAB ACTIVE
2916
2917 016636  004737  016714            350$:   JSR     PC,600$                 ;PRINT SPACES FOR TAB
2918 016642  000762                            BR      200$                    ;AND RETRIEVE NEXT CHARACTER
2919
2920                                   ;        LAST CHAR OR END OF FILE
2921
2922 016644  112704  000000            400$:   MOVB    #0,R4                   ;INSURE NULL CHAR
2923
2924 016650  004737  016714            425$:   JSR     PC,600$                 ;EMPTY BUFFER
2925 016654  112721  000015                    MOVB    #CR,(R1)+               ;OUTPUT CARRAGE RETURN
2926 016660  112721  000012                    MOVB    #LF,(R1)+               ;AND LINE FEED
2927 016664  112704  000000                    MOVB    #0,R4                   ;FOLLOWED BY NULL
2928 016670  004737  016714                    JSR     PC,600$                 ;PRINT
2929
2930                                   ;        CLOSE FILE AND RETURN TO MAIN
2931
2932 016674                            500$:   CLOSE                           ;CLOSE FILE
     016674  104435                            TRAP    C$CLOS
2933 016676  013701  002754                    MOV     REG1,R1                 ;RESTORE
2934 016702  013702  002756                    MOV     REG2,R2                 ;REGISTERS
2935 016706  013704  002762                    MOV     REG4,R4
2936 016712  000207                            RTS     PC                      ;AND RETURN
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBROUTINE TO TRANSFER CHARACTERS TO PRINT BUFFER AND DUMP
                                        ;;      BUFFER WHEN FULL.
                                        STARS
                                        ;;*****************************************************************

                                        ;       MOVE CHAR TO BUFFER, TEST FOR NULL CHAR OR END OF BUFFER

016714  110421          005350  600$:   MOVB    R4,(R1)+                ;MOVE CHAR TO BUFFER
016716  001403                          BEQ     700$                    ;PRINT BUFFER IF NULL
016720  022701                          CMP     #EOL,R1                 ;TEST FOR END OF BUFFER
016724  001012                          BNE     800$                    ;NO:  CONTINUE

                                        ;       PRINT BUFFER AND RESET POINTER

016726                          700$:   PRINTB  #LINE                   ;PRINT LINE AT CONSOLE
016726  012746  005256                  MOV     #LINE,-(SP)
016732  012746  000001                  MOV     #1,-(SP)
016736  010600                          MOV     SP,R0
016740  104414                          TRAP    C$PNTB
016742  062706  000004                  ADD     #4,SP
016746  012701  005260                  MOV     #LINBUF,R1              ;AND RESET POINTER

                                        ;       TEST FOR CONTROL CHARS, TAB ACTIVE OR RETURN

016752  120427  000040          800$:   CMPB    R4,#BLK                 ;COMPARE CHAR WITH BLANK
016756  100406                          BMI     850$                    ;BRANCH IF CONTROL CHAR

016760  005302                          DEC     R2                      ;DEC TAB COUNTER
016762  001404                          BEQ     850$                    ;BRANCH TO RESET
016764  005737  003020                  TST     TABFLG                  ;NOW TEST TAB FLAG
016770  001351                          BNE     600$                    ;ACTIVE:  MOVE ANOTHER BLANK
016772  000207                          RTS     PC                      ;NO:  JUST RETURN

                                        ;       RESET TAB COUNTER, TAB ACTIVE FLAG AND RETURN

016774  012702  000010          850$:   MOV     #8.,R2                  ;RESET TAB
017000  005037  003020                  CLR     TABFLG                  ;CLEAR TAB ACTIVE FLAG
017004  000207                          RTS     PC                      ;AND RETURN
```

```
                                .SBTTL  EXPAND BIT NAMES
017006                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  ROUTINE TO EXPAND REGISTER BIT NAMES
                                ;;
                                ;;  INPUT:              FP.BIT::        POINTER TO 72 BYTE PRINT BUFFER
                                ;;                      FP.TBL::        POINTER TO BIT EXPAND TABLE
                                ;;                      FP.XOR::        VALUE TO EXPAND
                                ;;
                                ;;  CALLING SEQUENCE:   JSR     PC,DSPBITS
017006                          STARS
                                ;;*****************************************************************

017006                          DSPBITS::

                                ;       SAVE REGISTERS R1,R2 & R4

017006  010137  002754                  MOV     R1,REG1         ;SAVE
017012  010237  002756                  MOV     R2,REG2         ;REGISTERS
017016  010437  002762                  MOV     R4,REG4

                                ;       INITIALIZE PARAMETERS

017022  005000                          CLR     R0                      ;CLEAR CHARACTER
017024  013701  002746                  MOV     FP.TBL,R1               ;LOAD EXPAND TABLE ADDRESS
017030  013702  002750                  MOV     FP.BIT,R2               ;LOAD PRINT BUFFER ADDRESS
017034  012704  100000                  MOV     #100000,R4              ;INITIALIZE BIT SELECT
017040  012737  000110  003006          MOV     #72.,BITCNT             ;AND INITIALIZE CHARACTER COUNT

017046  005000                          CLR     R0                      ;PRIME REGISTER WITH NULL
017050  005737  002740                  TST     FP.XOR                  ;TEST FOR NON ZERO VALUE
017054  001434                          BEQ     400$                    ;BRANCH IF ZERO
017056  000403                          BR      150$                    ;TEST FIRST BIT
```

```
3006
3007                                    ;       SHIFT BIT SELECT AND TEST
3008
3009 017060  000241                    100$:   CLC                             ;CLEAR CARRY BEFORE SHIFT
3010 017062  006004                            ROR     R4                      ;SHIFT SELECT TO NEXT BIT
3011 017064  001432                            BEQ     450$                    ;BRANCH IF SELECT REGISTER ZERO
3012
3013 017066  030437  002740          150$:   BIT     R4,FP.XOR               ;TEST FOR SELECT BIT SET
3014 017072  001006                            BNE     300$                    ;BRANCH IF BIT SET
3015
3016                                    ;       ADVANCE TABLE POINTER PAST NEXT DELIMETER
3017
3018 017074  112100                    200$:   MOVB    (R1)+,R0                ;READ CHAR FROM TABLE & BUMP PTR
3019 017076  001425                            BEQ     450$                    ;BRANCH IF NULL CHARACTER
3020
3021 017100  120027  000054            210$:   CMPB    R0,#54                  ;COMPARE WITH DELIMETER
3022 017104  001373                            BNE     200$                    ;BRANCH BACK IF NOT DELIMETER
3023 017106  000764                            BR      100$                    ;CONTINUE WITH NEXT BIT
3024
3025                                    ;       MOVE EXPANDED NAME TO PRINT BUFFER AND ADVANCE TABLE PTR
3026
3027 017110  112100                    300$:   MOVB    (R1)+,R0                ;READ CHAR FROM TABLE & BUMP PTR
3028 017112  001417                            BEQ     450$                    ;BRANCH IF NULL CHARACTER
3029
3030 017114  120027  000054            310$:   CMPB    R0,#54                  ;COMPARE WITH DELIMETER
3031 017120  001405                            BEQ     350$                    ;BRANCH IF DELIMETER
3032
3033 017122  110022                    325$:   MOVB    R0,(R2)+                ;MOVE CHAR TO BUFFER & BUMP PTR
3034 017124  005337  003006                    DEC     BITCNT                  ;DECREMENT CHARACTER COUNT
3035 017130  001367                            BNE     300$                    ;MOVE NEXT CHAR IF COUNT NOT ZERO
3036 017132  000405                            BR      400$                    ;COUNT EXPIRED:  ADD PROMPT CHAR
3037
3038 017134  112722  000040            350$:   MOVB    #40,(R2)+               ;MOVE BLANK CHAR TO BUFFER & BUMP PTR
3039 017140  005337  003006                    DEC     BITCNT                  ;DECREMENT CHARACTER COUNT
3040 017144  001345                            BNE     100$                    ;TEST NEXT BIT IF COUNT NOT ZERO
3041
3042                                    ;       CHARACTER COUNT EXPIRED:  MOVE PROMPT CHAR TO BUFFER
3043
3044 017146  112722  000077            400$:   MOVB    #77,(R2)+               ;MOVE PROMPT & BUMP PTR
```

EXPAND BIT NAMES

```
3046
3047                                    ;       LAST CHARACTER TO MOVE:  NULL
3048
3049 017152  110012                    450$:   MOVB    R0,(R2)                 ;MOVE NULL INTO BUFFER
3050
3051                                    ;       RESTORE REGISTERS R1,R2 & R4
3052
3053 017154  013701  002754            500$:   MOV     REG1,R1                 ;RESTORE
3054 017160  013702  002756                    MOV     REG2,R2                 ;REGISTERS
3055 017164  013704  002762                    MOV     REG4,R4
3056
3057 017170  000207                            RTS     PC                      ;AND RETURN
3058
3059 017172                                    ENDMOD
3060
3061                                    .SBTTL
```

```
                                .SBTTL  REPORT CODING SECTION

                                        BGNMOD

                                .SBTTL  REPORT CODING SECTION
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  REPORT SECTION GENERATES SUMMARY
                                ;;  AND CLEARS TABLES FOR EACH TEST EXECUTED.
                                ;;
                                ;;      INPUT   RP.SIZ          MAX NUMBER OF TESTS
                                ;;              RP.PASS::       PASS COUNT TABLE
                                ;;              RP.ERROR::      ERROR COUNT TABLE
                                ;;
                                ;;      OUTPUT                  TEST NO., PASS COUNT & ERROR COUNT
                                STARS
                                ;;*****************************************************************

                                        BGNRPT
017172                          L$RPT::

                                        PRINTS  #F.RPT1                 ;PRINT REPORT HEADER
017172  012746  014074                  MOV     #F.RPT1,-(SP)
017176  012746  000001                  MOV     #1,-(SP)
017202  010600                          MOV     SP,R0
017204  104416                          TRAP    C$PNTS
017206  062706  000004                  ADD     #4,SP

017212  012701  000001                  MOV     #1,R1                   ;INITIALIZE TEST NUMBER
017216  012702  000000                  MOV     #0,R2                   ;INITIALIZE TEST INDEX

017222  005762  002520          100$:   TST     RP.PASS(R2)             ;TEST FOR MORE THAN ONE PASS
017226  001415                          BEQ     200$                    ;CONTINUE IF NO PASS

                                        PRINTS  #F.RPT2,R1,RP.PASS(R2),RP.ERROR(R2)
017230  016246  002620                  MOV     RP.ERROR(R2),-(SP)
017234  016246  002520                  MOV     RP.PASS(R2),-(SP)
017240  010146                          MOV     R1,-(SP)
017242  012746  014164                  MOV     #F.RPT2,-(SP)
017246  012746  000004                  MOV     #4,-(SP)
017252  010600                          MOV     SP,R0
017254  104416                          TRAP    C$PNTS
017256  062706  000012                  ADD     #12,SP

017262  005062  002520          200$:   CLR     RP.PASS(R2)             ;CLEAR PASS COUNT
017266  005062  002620                  CLR     RP.ERROR(R2)            ;CLEAR ERROR COUNT

017272  005201                          INC     R1                      ;INCREMENT TEST NO.
017274  062702  000002                  ADD     #2,R2                   ;BUMP INDEX
017300  022701  000033                  CMP     #MAXTST,R1              ;COMPARE WITH LAST TEST NO
017304  100346                          BPL     100$                    ;AND CONTINUE UNTIL DONE

017306                                  EXIT    RPT
017306  000167                          .WORD   J$JMP
017310  000000                          .WORD   L10011-2-.
```

```
3140                                            .EVEN
3141
3142 017312                                     ENDRPT
     017312                             L10011:
     017312  104425                             TRAP    C$RPT
```

```
                                .SBTTL  PROTECTION TABLE
                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

017314                                  BGNPROT
017314                          L$PROT::

177777                                  -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
177777                                  -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
177777                                  -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

017322                                  ENDPROT
```

```
                                .SBTTL  PROGRAM INITIALIZATION SECTION
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  INITIALIZATION SECTION:
                                ;;
                                ;;  STEPS:      IF NOT INIFLG THEN:
                                ;;                  ASK FOR & PRINT SUPERVISOR HELP FILE
                                ;;                  ASK FOR & PRINT DIAGNOSTIC HELP FILE
                                ;;
                                ;;              RETRIEVE P-TABLE ADDRESS & SAVE DATA LOCALLY
                                ;;
                                ;;              IF PROGRAM START THEN:
                                ;;                  RESET UNIBUS
                                ;;                  CLEAR PRINT MSG FLAGS AND REPORT TABLES
                                ;;                  DETERMINE CPU TYPE & GENERATE REGISTER SELECT MASKS
                                ;;                  INITIALIZE REGISTER ADDRESS TABLE
                                ;;                  PROCESS INTERFACE DRIVE NO. & COMPUTE ATTN BIT POSITION
                                ;;
                                ;;              DETERMINE BOARD REV & INITIALIZE DRIVE TYPE
                                ;;              BUILD TEST SELECT MASK FROM SOFTWARE QUESTION DATA
                                ;;              ENABLE INTERRUPTS AND BEGIN TESTING
                                ;;
                                STARS
                                ;;*****************************************************************

                                        BGNINIT
                                L$INIT::
                                ;       PROCESS PROGRAM INITIALIZATION

017322  005737  003014          100$:   TST     INIFLG                  ;TEST FOR PROGRAM INITIALIZATION
017326  001042                          BNE     200$                    ;INITIALIZED:  CONTINUE

017330  005237  003014          120$:   INC     INIFLG                  ;SET INITIALIZATION FLAG
017334  005037  003016                  CLR     HLPFLG                  ;CLEAR OPERATOR HELP FLAG
017340                                  GMANIL  Q.HLP1,HLPFLG,1,YES     ;ASK OPERATOR FOR SUPERVISOR HELP
017340  104443                          TRAP    C$GMAN
017342  000404                          BR      10000$
017344  003016                          .WORD   HLPFLG
017346  000130                          .WORD   T$CODE
017350  014334                          .WORD   Q.HLP1
017352  000001                          .WORD   1
017354                          10000$:
017354  005737  003016                  TST     HLPFLG                  ;TEST RESPONSE
017360  001405                          BEQ     150$                    ;NO:  CONTINUE

017362  012737  014371  002504          MOV     #D.FIL1,D.FILPTR        ;LOAD FILE NAME PTR
017370  004737  016536                  JSR     PC,DSPFILE              ;DISPLAY FILE

017374  005037  003016          150$:   CLR     HLPFLG                  ;CLEAR FLAG
017400                                  GMANIL  Q.HLP2,HLPFLG,1,YES     ;ASK OPERATOR FOR DIAGNOSTIC HELP
017400  104443                          TRAP    C$GMAN
017402  000404                          BR      10001$
017404  003016                          .WORD   HLPFLG
017406  000130                          .WORD   T$CODE
017410  014402                          .WORD   Q.HLP2
017412  000001                          .WORD   1
```

```
        017414                                  10001$:
3199    017414  005737  003016                          TST     HLPFLG                  ;TEST RESPONSE
3200    017420  001405                                  BEQ     200$                    ;NO:  CONTINUE
3201
3202    017422  012737  014437  002504                 MOV     #D.FIL2,D.FILPTR        ;LOAD FILE NAME PTR
3203    017430  004737  016536                          JSR     PC,DSPFILE              ;DISPLAY FILE
```

```
3205 017434                            STARS
                                       ;;*****************************************************************
3206                                   ;;
3207                                   ;;      RETRIEVE P-TABLE ADDRESS FOR EACH NEW PASS
3208 017434                            STARS
                                       ;;*****************************************************************
3209
3210 017434                            200$:   GPHARD  LOGUNIT,HRDTBL          ;RETRIEVE P-TABLE ADDRESS
     017434  013700  002774                    MOV     LOGUNIT,R0
     017440  104442                            TRAP    C$GPHRD
     017442  010037  002772                    MOV     R0,HRDTBL
3211
3212 017446                                    BCOMPLETE 220$                  ;CONTINUE IF SUCCESSFUL
     017446  103413                            BCS     220$
3213
3214 017450                                    PRINTF  #F.UNIT,LOGUNIT         ;REPORT P-TABLE ERROR
     017450  013746  002774                    MOV     LOGUNIT,-(SP)
     017454  012746  014036                    MOV     #F.UNIT,-(SP)
     017460  012746  000002                    MOV     #2,-(SP)
     017464  010600                            MOV     SP,R0
     017466  104417                            TRAP    C$PNTF
     017470  062706  000006                    ADD     #6,SP
3215
3216 017474                                    DOCLN                           ;ABORT THIS PASS
     017474  104444                            TRAP    C$DCLN
3217
3218 017476  013701  002772            220$:   MOV     HRDTBL,R1               ;LOAD TABLE ADDRESS
3219 017502  012137  002720                    MOV     (R1)+,PT.CSR            ;AND MOVE
3220 017506  012137  002722                    MOV     (R1)+,PT.VEC            ;PARAMETERS
3221 017512  012137  002724                    MOV     (R1)+,PT.PRI            ;TO LOCAL
3222 017516  012137  002726                    MOV     (R1)+,PT.DRI            ;STORAGE
```

```
3224 017522                            STARS
                                       ;;*****************************************************************
3225                                   ;;
3226                                   ;;      TEST AND PROCESS START CONDITION
3227 017522                            STARS
                                       ;;*****************************************************************
3228
3229 017522                            300$:   READEF  #EF.START               ;TEST FOR OPERATOR START
     017522  012700  000040                    MOV     #EF.START,R0
     017526  104447                            TRAP    C$REFG
3230
3231 017530                                    BNCOMPLETE 400$                 ;CONTINUE IF NOT
     017530  103150                            BCC     400$
3232
3233 017532                                    BRESET                          ;BUS RESET
     017532  104433                            TRAP    C$RESET
3234
3235                                   ;       CLEAR PRINT FLAGS
3236
3237 017534  005037  003042                    CLR     CSRFLG                  ;ENABLE CSR ADDRESS MSG
3238 017540  005037  003044                    CLR     NEDFLG                  ;ENABLE DRIVE SELECT MSG
3239 017544  005037  003046                    CLR     PWRFLG                  ;ENABLE POWER ON MSG
3240 017550  005037  003050                    CLR     CBLFLG                  ;ENABLE CABLE MSG
3241 017554  005037  003052                    CLR     PERFLG                  ;ENABLE PARITY MSG
3242
3243                                   ;       INITIALIZE REPORT TABLES
3244
3245 017560  005000                            CLR     R0                      ;CLEAR INDEX REGISTER
3246
3247 017562  005060  002520            310$:   CLR     RP.PASS(R0)             ;CLEAR PASS COUNT
3248 017566  005060  002620                    CLR     RP.ERROR(R0)            ;CLEAR ERROR COUNT
3249 017572  062700  000002                    ADD     #2,R0                   ;BUMP INDEX
3250 017576  022700  000100                    CMP     #RP.SIZ,R0              ;COMPARE WITH MAX TEST INDEX
3251 017602  003367                            BGT     310$                    ;CONTINUE UNTIL TABLES CLEAR
3252
3253                                   ;       DETERMINE PROCESSOR & MASSBUS CONTROLLER TYPE
3254
3255 017604                            320$:   SETVEC  #4,#TRAP4,#PRI07        ;SETUP ADDRESS TRAP SERVICE
     017604  012746  000340                    MOV     #PRI07,-(SP)
     017610  012746  015220                    MOV     #TRAP4,-(SP)
     017614  012746  000004                    MOV     #4,-(SP)
     017620  012746  000003                    MOV     #3,-(SP)
     017624  104437                            TRAP    C$SVEC
     017626  062706  000010                    ADD     #10,SP
3256
3257 017632  005037  003024                    CLR     TRPFLG                  ;CLEAR TRAP FLAG
3258 017636  013700  177764                    MOV     CPUID,R0                ;TEST PDP-11/70 LOCATION
3259 017642  005737  003024                    TST     TRPFLG                  ;TEST FOR ADDRESS TRAP
3260 017646  001002                            BNE     340$                    ;NO TRAP:  CONTINUE
3261
3262 017650  005237  003012                    INC     CPUFLG                  ;SET PDP-11/70 FLAG
3263
3264 017654                            340$:   CLRVEC  #4                      ;CLEAR TRAP SERVICE
     017654  012700  000004                    MOV     #4,R0
     017660  104436                            TRAP    C$CVEC
3265
3266                                   ;       GENERATE REGISTER SELECTION MASKS
```

```
3267
3268 017662  012737  000024  002466  350$:   MOV     #24,R.MAX               ;INITIALIZE MAX REGISTER INDEX
3269 017670  012737  001777  002474          MOV     #1777,R.SAD             ;INITIALIZE REGISTER ADDRESS SELECT
3270 017676  012737  001271  002476          MOV     #1271,R.SIN             ;INITIALIZE INITIAL VALUE SELECT
3271 017704  012737  000027  002500          MOV     #27,R.SRH               ;INITIALIZE CONTROLLER REG SELECT
3272 017712  012737  001750  002502          MOV     #1750,R.SDR             ;INITIALIZE INTERFACE REG SELECT
3273
3274 017720  005737  003012                  TST     CPUFLG                  ;TEST FOR PDP-11/70 CPU
3275 017724  001411                          BEQ     360$                    ;NO:  CONTINUE
3276
3277 017726  012737  000030  002466          MOV     #30,R.MAX               ;ADD TWO REGISTERS TO MAX INDEX
3278 017734  012737  007777  002474          MOV     #7777,R.SAD             ;ADD TWO REGISTERS TO ADDRESS SELECT
3279 017742  012737  006027  002500          MOV     #6027,R.SRH             ;ADD TWO REGISTERS TO CNTRLLR SELECT
3280
3281                                 ;       INITIALIZE REGISTER ADDRESS TABLE
3282
3283 017750  013701  002720          360$:   MOV     PT.CSR,R1               ;LOAD UNIBUS BASE ADDRESS
3284 017754  005002                          CLR     R2                      ;AND CLEAR INDEX REGISTER
3285
3286 017756  010104                  370$:   MOV     R1,R4                   ;LOAD BASE ADDRESS
3287 017760  066204  002246                  ADD     D.TBL(R2),R4            ;ADD OFFSET FROM TABLE
3288 017764  010462  002276                  MOV     R4,A.TBL(R2)            ;AND STORE INTO TABLE
3289 017770  062702  000002                  ADD     #2,R2                   ;BUMP REGISTER INDEX
3290 017774  020237  002466                  CMP     R2,R.MAX                ;COMPARE WITH MAX INDEX
3291 020000  002766                          BLT     370$                    ;AND CONTINUE IF LESS THAN MAX
3292
3293                                 ;       PROCESS INTERFACE DRIVE NO. & COMPUTE ATTN BIT POSITION
3294
3295 020002  013701  002726          380$:   MOV     PT.DRI,R1               ;LOAD DRIVE SELECT NUMBER
3296 020006  042737  000007  002366          BIC     #7,IN.CS2               ;CLEAR DRIVE SELECT IN CS2 INITIAL VAL
3297 020014  050137  002366                  BIS     R1,IN.CS2               ;SET DRIVE SELECT BITS ONLY
3298 020020  010137  003000                  MOV     R1,DRIVE                ;SAVE DRIVE SELECT
3299
3300 020024  012737  000001  002776          MOV     #BIT00,UNITPOS          ;INITIALIZE ATTN BIT POSITION
3301 020032  013701  003000                  MOV     DRIVE,R1                ;RELOAD DRIVE NO.
3302 020036  001423                          BEQ     420$                    ;STOP NOW IF DRIVE ZERO
3303
3304 020040  006337  002776          390$:   ASL     UNITPOS                 ;SHIFT TO NEXT POSITION
3305 020044  005301                          DEC     R1                      ;DECREMENT UNIT NUMBER
3306 020046  001374                          BNE     390$                    ;CONTINUE UNTIL ZERO
3307 020050  000416                          BR      420$                    ;PROCESS RESTART
```

```
3309 020052                            STARS
                                       ;;*****************************************************************
3310                                   ;;
3311                                   ;;      COMMON INITIALIZATION SECTIONS
3312 020052                            STARS
                                       ;;*****************************************************************
3313
3314                                   ;       INITIALIZE REPORT TABLES
3315
3316 020052                            400$:   READEF  #EF.RESTART             ;TEST FOR OPERATOR RESTART
     020052  012700  000037                    MOV     #EF.RESTART,R0
     020056  104447                            TRAP    C$REFG
3317
3318 020060                                    BNCOMPLETE 440$                 ;CONTINUE IF NOT
     020060  103022                            BCC     440$
3319
3320 020062  005000                            CLR     R0                      ;CLEAR INDEX REGISTER
3321
3322 020064  005060  002520            410$:   CLR     RP.PASS(R0)             ;CLEAR PASS COUNT
3323 020070  005060  002620                    CLR     RP.ERROR(R0)            ;CLEAR ERROR COUNT
3324 020074  062700  000002                    ADD     #2,R0                   ;BUMP INDEX
3325 020100  022700  000100                    CMP     #RP.SIZ,R0              ;COMPARE WITH MAX TEST INDEX
3326 020104  003367                            BGT     410$                    ;CONTINUE UNTIL TABLES CLEAR
3327
3328                                   ;       PROCESS BOARD REVISION LEVEL
3329
3330 020106  012737  000005  002400  420$:   MOV     #5,IN.DT                ;REV B DRIVE TYPE
3331 020114  005737  002226                    TST     REVFLG                  ;TEST FOR REV A BOARD
3332 020120  001402                            BEQ     440$                    ;NO:  CONTINUE
3333 020122  005337  002400                    DEC     IN.DT                   ;DECREMENT DRIVE TYPE FOR REV A
3334
3335                                   ;       BUILD TEST SELECT WORD:  BITS 0-7
3336
3337                                   ;       : REVA :  NU  :  NU  : UPAR : BYTE : ABOR : AT03 : AT00 :
3338
3339 020126  005001                    440$:   CLR     R1                      ;CLEAR SELECT
3340 020130  053701  002226                    BIS     REVFLG,R1               ;SET/RESET REV "A" BIT
3341 020134  006301                            ASL     R1                      ;SHIFT PAST
3342 020136  006301                            ASL     R1                      ;UNUSED BIT POSITIONS
3343 020140  006301                            ASL     R1                      ;AND SHIFT SELECT
3344 020142  053701  002240                    BIS     PARFLG,R1               ;SET/RESET PARITY ERROR BIT
3345 020146  006301                            ASL     R1                      ;AND SHIFT SELECT
3346 020150  053701  002236                    BIS     BYTFLG,R1               ;SET/RESET BYTE MODE XFER BIT
3347 020154  006301                            ASL     R1                      ;AND SHIFT SELECT
3348 020156  053701  002234                    BIS     ABOFLG,R1               ;SET/RESET XFER ABORT BIT
3349 020162  006301                            ASL     R1                      ;AND SHIFT SELECT
3350 020164  053701  002232                    BIS     AT3FLG,R1               ;SET/RESET AT3 ATTN BIT
3351 020170  006301                            ASL     R1                      ;AND SHIFT SELECT
3352 020172  053701  002230                    BIS     AT0FLG,R1               ;SET/RESET AT0 ATTN BIT
3353 020176  010137  003012                    MOV     R1,SELECT               ;AND SAVE SELECT
3354
3355                                   :       ENABLE INTERRUPTS AND BEGIN TESTING
3356
3357 020202                            500$:   SETPRI  PRI00                   ;ENABLE INTERRUPTS
     020202  013700  000000                    MOV     PRI00,R0
     020206  104441                            TRAP    C$SPRI
3358
```

```
 3359 020210                                ENDINIT
      020210                        L10013:
      020210  104411                       TRAP    C$INIT
```

```
                                .SBTTL  AUTODROP SECTION
                                ;++
                                ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                                ; THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                                ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                                ; DROPPED FROM TESTING.
                                ;--

020212                                  BGNAUTO
020212                          L$AUTO::

020212                                  ENDAUTO
020212                          L10014:
020212  104461                          TRAP    C$AUTO
```

```
                                .SBTTL  CLEANUP CODING SECTION100
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;      CLEANUP CODING SECTION:
                                ;;
                                ;;              (1)     DISABLE INTERRUPTS
                                ;;              (2)     RESTORE DEVICE INTERRUPT SERVICE VECTOR
                                STARS
                                ;;*****************************************************************

                                        BGNCLN
                                L$CLEAN::

                                        SETPRI  PRI07           ;DISABLE INTERRUPTS
020214  013700  000340                  MOV     PRI07,R0
020220  104441                          TRAP    C$SPRI

                                        EXIT    CLN
020222  104432                          TRAP    C$EXIT
020224  000002                          .WORD   L10015-.

                                        .EVEN

                                        ENDCLN
020226                          L10015:
020226  104412                          TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION
                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO NO LONGER BE TESTED.
                                ;--

020230                                  BGNDU
020230                          L$DU::
020230  000240                          NOP                     ;DROP UNIT NOT IMPLEMENTED

020232                                  ENDDU
020232                          L10016:
020232  104453                          TRAP    C$DU
```

```
                                .SBTTL  ADD UNIT SECTION
                                ;++
                                ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
                                ;  TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
                                ;  TO THE TEST CYCLE.
                                ;--

020234                                  BGNAU
020234                          L$AU::

020234  000240                            NOP                     ;ADD UNIT NOT IMPLEMENTED

020236                                  ENDAU
020236                          L10017:
020236  104452                          TRAP    C$AU

020240                                  ENDMOD

                                .SBTTL
```

```
   3460 020240                                  BGNMOD
```

```
                                .SBTTL  TEST  1:  REGISTER ADDRESSING
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK DEVICE REGISTER ADDRESSING
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      EXECUTE SUBTEST 1-2
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MEMORY ADDRESS TRAP ON REGISTER ACCESS
                                STARS
                                ;;*****************************************************************

                                        BGNTST                          ;REGISTER ADDRESSING
020240                          T1::
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      SET UP BUS ADDRESS TRAP SERVICE
                                        ;;      READ REGISTER AND TEST FOR UNIBUS TRAP
                                        ;;      CLEAR BUS ADDRESS TRAP VECTOR
                                        ;;      PROCESS ANY ADDRESS ERROR
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB                          ;UNIBUS ADDRESS TRAP
                                        T1.1:
                                                TRAP    C$BSUB

                                                JSR     PC,SETRAP               ;SETUP ADDRESS TRAP SERVICE

                                        100$:   CLR     TRPFLG                  ;CLEAR TRAP FLAG
                                                CLR     R2                      ;CLEAR TABLE INDEX
                                                CLR     R.ERR                   ;CLEAR ERROR FLAG
                                                MOV     #1,R4                   ;LOAD REGISTER ADDR SELECT BIT

                                        200$:   BIT     R4,R.SAD                ;TEST THIS REGISTER?
                                                BEQ     300$                    ;NO:  MOVE TO NEXT

                                                TST     @A.TBL(R2)              ;ADDRESS REGISTER
                                                TST     TRPFLG                  ;TEST FOR TIME-OUT
                                                BEQ     300$                    ;NO:  CONTINUE

                                                CLR     TRPFLG                  ;YES:  CLEAR FLAG
                                                BIS     R4,R.ERR                ;SET REGISTER ADDR ERROR BIT

                                        300$:   ASL     R4                      ;ADJUST SELECT BIT
                                                ADD     #2,R2                   ;BUMP INDEX REGISTER
                                                CMP     R2,R.MAX                ;COMPARE INDEX TO TABLE SIZE
                                                BLT     200$                    ;AND CONTINUE IF NOT DONE

                                        400$:   JSR     PC,CLRTRAP              ;CLEAR TRAP VECTOR
                                                TST     R.ERR                   ;TEST FOR ERRORS
                                                BEQ     500$                    ;FLAG ZERO:  CONTINUE

                                                ERRDF   1,M.ADR1,ER.ADR         ;"DEVICE ADDRESS ERROR"
                                                TRAP    C$ERDF
                                                .WORD   1
                                                .WORD   M.ADR1
                                                .WORD   ER.ADR

                                                TST     CSRFLG                  ;TEST PRINT FLAG
                                                BNE     450$                    ;CONTINUE IF SET

                                                PRINTF  #F.CSR                  ;PRINT CSR ADDRESS MSG
                                                MOV     #F.CSR,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTF
                                                ADD     #4,SP
```

```
3524 020400  005237  003042                    INC     CSRFLG                  ;SET FLAG
3525
3526 020404                            450$:   DOCLN                           ;CLEAN UP AND EXIT
     020404  104444                            TRAP    C$DCLN
3527
3528 020406                            500$:   ENDSUB                          ;UNIBUS ADDRESS TRAP
     020406                            L10021:
     020406  104403                            TRAP    C$ESUB
```

```
3530 020410                            STARS
                                       ;;*****************************************************************
3531                                   ;;
3532                                   ;;    SUBTEST 2:
3533                                   ;;
3534                                   ;;      READ INTERFACE REGISTER
3535                                   ;;      TEST FOR NON-EXISTENT DRIVE STATUS
3536                                   ;;      PROCESS ANY ADDRESS ERROR(S)
3537                                   ;;
3538                                   ;;      ISSUE MASSBUS CONTROLLER CLEAR AND TEST FOR ERROR
3539 020410                            STARS
                                       ;;*****************************************************************
3540
3541 020410                                    BGNSUB
     020410                            T1.2:
     020410  104402                            TRAP    C$BSUB
3542
3543 020412  005002                    100$:   CLR     R2                      ;CLEAR TABLE INDEX
3544 020414  005037  002470                    CLR     R.ERR                   ;CLEAR ERROR FLAG
3545 020420  005037  003026                    CLR     ERRFLG                  ;CLEAR FLAG
3546 020424  012704  000001                    MOV     #1,R4                   ;LOAD REGISTER ADDR SELECT BIT
3547
3548 020430  030437  002502            200$:   BIT     R4,R.SDR                ;TEST THIS REGISTER?
3549 020434  001412                            BEQ     300$                    ;NO:  MOVE TO NEXT
3550
3551 020436  004737  015342                    JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
3552
3553 020442  005772  002276                    TST     @A.TBL(R2)              ;ADDRESS REGISTER
3554 020446  032777  010000  161632            BIT     #CS2.NED,@A.CS2         ;TEST FOR NON-EXISTENT DEVICE
3555 020454  001402                            BEQ     300$                    ;NO:  CONTINUE
3556
3557 020456  050437  002470                    BIS     R4,R.ERR                ;SET REGISTER ADDR ERROR BIT
3558
3559 020462  006304                    300$:   ASL     R4                      ;ADJUST SELECT BIT
3560 020464  062702  000002                    ADD     #2,R2                   ;BUMP INDEX REGISTER
3561 020470  020237  002466                    CMP     R2,R.MAX                ;COMPARE INDEX TO TABLE SIZE
3562 020474  002755                            BLT     200$                    ;AND CONTINUE IF NOT DONE
3563
3564 020476  005737  002470            400$:   TST     R.ERR                   ;TEST FOR ERRORS
3565 020502  001421                            BEQ     600$                    ;FLAG ZERO:  CONTINUE
3566
3567 020504  005237  003026                    INC     ERRFLG                  ;SET FLAG FOR LATER
3568 020510                                    ERRDF   2,M.ADR2,ER.ADR         ;"NON-EXISTENT DEVICE STATUS"
     020510  104455                            TRAP    C$ERDF
     020512  000002                            .WORD   2
     020514  006263                            .WORD   M.ADR2
     020516  014452                            .WORD   ER.ADR
3569
3570 020520                                    PRINTX  #F.NED                  ;CHECK DRIVE SELECT SWITCHES 1-3
     020520  012746  013732                    MOV     #F.NED,-(SP)
     020524  012746  000001                    MOV     #1,-(SP)
     020530  010600                            MOV     SP,R0
     020532  104415                            TRAP    C$PNTX
     020534  062706  000004                    ADD     #4,SP
3571 020540  005237  003044                    INC     NEDFLG                  ;SET FLAG
3572
3573 020544                            500$:   DOCLN                           ;CLEAN UP AND EXIT
```

```
        020544  104444                          TRAP    C$DCLN
  3574
  3575  020546                          600$:   ENDSUB                  ;NON-EXISTENT DEVICE
        020546                          L10022:
        020546  104403                          TRAP    C$ESUB
  3576
  3577  020550  004737  015550                  JSR     PC,CLRTST       ;AND CLEAR TEST DATA
  3578
  3579  020554                                  ENDTST
        020554                          L10020:
        020554  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  2:  REGISTER INITIALIZATION
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      TEST REGISTER INITIALIZATION
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      INITIALIZE THE MASSBUS CONTROLLER
                                ;;      READ REGISTER AND TEST FOR EXPECTED VALUE
                                ;;      PROCESS ANY INITIAL VALUE ERRORS
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      UNEXPECTED/IMPROPER REGISTER VALUE
                                STARS
                                ;;*****************************************************************

                                        BGNTST                          ;REGISTER INITIALIZATION
                                T2::
                                        JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
                                        TST     MBCFLG                  ;TEST CLEAR FLAG
                                        BEQ     100$                    ;CONTINUE IF ZERO

                                        ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAr ERROR''
                                        TRAP    C$ERDF
                                        .WORD   1
                                        .WORD   M.MBC1
                                        .WORD   ER.DMP
                                        DOCLN                           ;CLEAN UP AND EXIT
                                        TRAP    C$DCLN

                                100$:   JSR     PC,RDREG                ;READ REGISTERS INTO R.TBL
                                        CLR     R2                      ;CLEAR TABLE INDEX
                                        CLR     R.ERR                   ;CLEAR ERROR FLAG
                                        MOV     #1,R4                   ;LOAD REGISTER INIT SELECT BIT

                                200$:   BIT     R4,R.SIN                ;TEST SELECTION BIT
                                        BEQ     300$                    ;NO:  MOVE TO NEXT REGISTER

                                        CMP     R.TBL(R2),IN.TBL(R2)    ;COMPARE WITH EXPECTED INIT VALUE
                                        BEQ     300$                    ;SAME:  CONTINUE

                                        BIS     R4,R.ERR                ;SET CORRESPONDING REGISTER ERROR BIT

                                300$:   ASL     R4                      ;SHIFT REGISTER SELECT BIT
                                        ADD     #2,R2                   ;BUMP TABLE INDEX
                                        CMP     R2,R.MAX                ;COMPARE WITH MAXIMUM REGISTER INDEX
                                        BLT     200$                    ;NO:  CONTINUE
```

```
3626 020656  005737  002470          400$:   TST     R.ERR                   ;TEST ERROR FLAG
3627 020662  001404                          BEQ     500$                    ;BRANCH IF NO ERRORS
3628
3629 020664                                   ERRHRD  2,M.REG1,ER.INI         ;"REGISTER INITIALIZATION ERROR"
     020664  104456                          TRAP    C$ERHRD
     020666  000002                          .WORD   2
     020670  012076                          .WORD   M.REG1
     020672  014640                          .WORD   ER.INI
3630
3631 020674  004737  015550          500$:   JSR     PC,CLRTST               ;AND CLEAR TEST DATA
3632
3633 020700                                   ENDTST                          ;REGISTER INITIALIZATION
     020700                          L10023:
     020700  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  3:  CHECK CONTROL/STATUS 1 FUNCTION BITS
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      TEST CS1 REGISTER FOR FUNCTION BITS F4->F1
                                ;;      STUCK AT ONE, ZERO OR STUCK TOGETHER
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR STATUS
                                ;;      EXECUTE SUB TESTS 1-4
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER CLEAR ERROR
                                ;;      CS1 REGISTER BITS STUCK
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T3::
020702  004737  015342                  JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
020706  005737  003030                  TST     MBCFLG                  ;TEST CLEAR FLAG
020712  001405                          BEQ     100$                    ;CONTINUE IF ZERO

020714                                  ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
020714  104455                          TRAP    C$ERDF
020716  000001                          .WORD   1
020720  011650                          .WORD   M.MBC1
020722  015074                          .WORD   ER.DMP
020724                                  DOCLN                           ;CLEAN UP AND EXIT
020724  104444                          TRAP    C$DCLN

020726  013737  002406  002730  100$:   MOV     NA.CS1,FP.NAM           ;LOAD REGISTER NAME ADDRESS
020734  013737  002436  002746          MOV     BA.CS1,FP.TBL           ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUB TESTS
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 1:
                                        ;;
                                        ;;        LOAD CS1 REGISTER FUNCTION BITS F4->F1 WITH ONE'S
                                        ;;        READ REGISTER AND TEST FOR BITS STUCK AT ZERO
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;LOAD ALL ONE'S
                                        T3.1:
                                                TRAP    C$BSUB

                                100$:   MOVB    #76,@A.CS1                      ;SET ALL FUNCTION BITS
                                        MOV     #4276,FP.EXP                    ;LOAD INIT EXPECTED CONTENTS
                                        MOV     @A.CS1,FP.ACT                   ;READ CS1 REGISTER
                                        CMP     FP.ACT,FP.EXP                   ;AND COMPARE WITH EXPECTED VALUE
                                        BEQ     400$                            ;IDENTICAL:  CONTINUE

                                        ERRHRD  2,M.CSR1,ER.REG                 ;"CS1 FUNCTION BIT ERROR"
                                        TRAP    C$ERHRD
                                        .WORD   2
                                        .WORD   M.CSR1
                                        .WORD   ER.REG

                                400$:   ENDSUB                                  ;ALL ONE'S
                                L10025:
                                        TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 2:
                                        ;;
                                        ;;        LOAD CS1 REGISTER FUNCTION BITS F4->F1 WITH ZERO'S
                                        ;;        READ REGISTER AND TEST FOR BITS STUCK AT ONE
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;LOAD ALL ZERO'S
                                        T3.2:
                                                TRAP    C$BSUB
                                        100$:   MOVB    #0,@A.CS1               ;WRITE TO CS1 REGISTER LSB
                                                MOV     #4200,FP.EXP            ;LOAD INIT EXPECTED VAL
                                                MOV     @A.CS1,FP.ACT           ;READ CS1 REGISTER
                                                CMP     FP.ACT,FP.EXP           ;AND COMPARE WITH EXPECTED VALUE
                                                BEQ     400$                    ;IDENTICAL:  CONTINUE

                                                ERRHRD  3,M.CSR1,ER.REG         ;''CS1 FUNCTION BIT ERROR''
                                                TRAP    C$ERHRD
                                                .WORD   3
                                                .WORD   M.CSR1
                                                .WORD   ER.REG

                                        400$:   ENDSUB                          ;LOAD ZERO'S
                                        L10026:
                                                TRAP    C$ESUB
```

```
021056                                STARS
                                      ;;*************************************************************
                                      ;;
                                      ;;      SUBTEST 3:
                                      ;;
                                      ;;        LOAD CS1 REGISTER FUNCTION BITS WITH MOVING ONE BIT
                                      ;;        READ REGISTER AND TEST FOR BITS STUCK AT ZERO
021056                                STARS
                                      ;;*************************************************************
021056                                        BGNSUB                          ;LOAD MOVING ONE
021056                                T3.3:
021056  104402                                TRAP    C$BSUB

021060  005037  002470                100$:   CLR     R.ERR                   ;CLEAR ERROR FLAG
021064  012702  000002                        MOV     #CS1.F0,R2              ;INITIALIZE CS1 BIT F0
021070  012705  000005                        MOV     #5,R5                   ;INITIALIZE BIT COUNTER

021074  110277  161176                200$:   MOVB    R2,@A.CS1               ;WRITE TO CS1 REGISTER LSB
021100  012737  004200  002734                MOV     #4200,FP.EXP            ;LOAD CS1 INIT EXPECTED CONTENTS
021106  060237  002734                        ADD     R2,FP.EXP               ;SET FUNCTION BIT
021112  017737  161160  002736        250$:   MOV     @A.CS1,FP.ACT           ;READ REGISTER
021120  023737  002736  002734                CMP     FP.ACT,FP.EXP           ;AND COMPARE BIT PATTERNS
021126  001423                                BEQ     400$                    ;IDENTICAL:  CONTINUE

021130  005737  002470                        TST     R.ERR                   ;TEST ERROR FLAG
021134  001016                                BNE     300$                    ;BRANCH IF NOT ZERO

021136  005237  002470                        INC     R.ERR                   ;INCREMENT ERROR FLAG
021142                                        ERRHRD  4,M.CSR1                ;"CS1 FUNCTION BIT ERROR"
021142  104456                                TRAP    C$ERHRD
021144  000004                                .WORD   4
021146  007210                                .WORD   M.CSR1
021150  000000                                .WORD   0
021152                                        PRINTX  #F.REG1                 ;PRINT REGISTER HEADER
021152  012746  013056                        MOV     #F.REG1,-(SP)
021156  012746  000001                        MOV     #1,-(SP)
021162  010600                                MOV     SP,R0
021164  104415                                TRAP    C$PNTX
021166  062706  000004                        ADD     #4,SP

021172  004737  016054                300$:   JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS

021176  006302                        400$:   ASL     R2                      ;AND SHIFT TO NEXT POSITION
021200  005305                                DEC     R5                      ;DECREMENT COUNTER
021202  001334                                BNE     200$                    ;CONTINUE WITH NEXT BIT POSITION

021204  005737  002470                500$:   TST     R.ERR                   ;TEST FOR ANY ERRORS
021210  001410                                BEQ     600$                    ;NO:  EXIT TEST
021212                                        PRINTX  #F.CRLF                 ;NEW LINE
021212  012746  014214                        MOV     #F.CRLF,-(SP)
021216  012746  000001                        MOV     #1,-(SP)
021222  010600                                MOV     SP,R0
021224  104415                                TRAP    C$PNTX
021226  062706  000004                        ADD     #4,SP

021232                                600$:   ENDSUB
021232                                L10027:
```

```
ZDRMAO LR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 110-1
TEST  3:  CHECK CONTROL/STATUS 1 FUNCTION BITS

        021232  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 4:
                                        ;;
                                        ;;
                                        ;;        LOAD CS1 REGISTER FUNCTION BITS WITH MOVING ZERO BIT
                                        ;;        READ REGISTER AND TEST FOR BITS STUCK AT ONE
                                        ;;
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;MOVING ZERO
                                        T3.4:
                                                TRAP    C$BSUB

                                        100$:   CLR     R.ERR                   ;CLEAR ERROR FLAG
                                                MOV     #74,R2                  ;INITIALIZE CS1 BITS F4->F1
                                                MOV     #5,R5                   ;INITIALIZE BIT COUNTER

                                        200$:   MOVB    R2,@A.CS1               ;WRITE TO CS1 REGISTER LSB
                                                MOV     #4200,FP.EXP            ;LOAD CS1 INIT EXPECTED CONTENTS
                                                ADD     R2,FP.EXP               ;SET/RESET FUNCTION BITS
                                        250$:   MOV     @A.CS1,FP.ACT           ;READ REGISTER
                                                CMP     FP.ACT,FP.EXP           ;AND COMPARE BIT PATTERNS
                                                BEQ     400$                    ;IDENTICAL:  CONTINUE

                                                TST     R.ERR                   ;TEST ERROR FLAG
                                                BNE     300$                    ;BRANCH IF NOT ZERO

                                                INC     R.ERR                   ;INCREMENT ERROR FLAG
                                                ERRHRD  5,M.CSR1                ;"CS1 FUNCTION BIT ERROR"
                                                TRAP    C$ERHRD
                                                .WORD   5
                                                .WORD   M.CSR1
                                                .WORD   0
                                                PRINTX  #F.REG1                 ;PRINT REGISTER HEADER
                                                MOV     #F.REG1,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
                                                ADD     #4,SP

                                        300$:   JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS

                                        400$:   ASL     R2                      ;MOVE ZERO BIT TO NEXT POSITION
                                                BIC     #CS1.IE,R2              ;CLEAR HIGH ORDER BIT
                                                BIS     #CS1.F0,R2              ;AND SET INCOMING FUNCTION BIT

                                                DEC     R5                      ;DECREMENT COUNTER
                                                BNE     200$                    ;CONTINUE WITH NEXT BIT POSITION

                                        500$:   TST     R.ERR                   ;TEST FOR ANY ERRORS
                                                BEQ     600$                    ;NO:  EXIT TEST
                                                PRINTX  #F.CRLF                 ;NEW LINE
                                                MOV     #F.CRLF,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
                                                ADD     #4,SP
```

```
ZDRMAO DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 111-1
TEST  3:  CHECK CONTROL/STATUS 1 FUNCTION BITS

   3782
   3783 021420                          600$:   ENDSUB
        021420                          L10030:
        021420  104403                          TRAP    C$ESUB
   3784
   3785 021422  004737  015550                 JSR     PC,CLRTST               ;AND CLEAR TEST DATA
   3786
   3787 021426                                  ENDTST
        021426                          L10024:
        021426  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  4:  CHECK INTERFACE FUNCTION/STATUS BITS
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK FS REGISTER FOR FUNCTION BITS F7->F0
                                ;;      STUCK AT ONE, ZERO OR STUCK TOGETHER
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      EXECUTE SUB TESTS 1-4
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      FUNCTION/STATUS BITS STUCK
                                ;;      LOOP-BACK CABLE NOT INSTALLED
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T4::
                                        JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
                                        TST     MBCFLG                  ;TEST CLEAR FLAG
                                        BEQ     100$                    ;CONTINUE IF ZERO

                                        ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
                                        TRAP    C$ERDF
                                        .WORD   1
                                        .WORD   M.MBC1
                                        .WORD   ER.DMP
                                        DOCLN                           ;CLEAN UP AND EXIT
                                        TRAP    C$DCLN

                                100$:   MOV     NA.FS,FP.NAM            ;LOAD FS REGISTER NAME ADDRESS
                                        MOV     BA.FS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST
```

```
021470                                   STARS
                                         ;;******************************************************************
                                         ;;
                                         ;;     SUBTEST 1:
                                         ;;
                                         ;;       LOAD FS REGISTER BITS F7-F0 WITH ONE'S.
                                         ;;       READ REGISTER CONTENTS AND CHECK FOR
                                         ;;       FUNCTION/STATUS BITS STUCK AT ZERO.
021470                                   STARS
                                         ;;******************************************************************
021470                                           BGNSUB                          ;LOAD ONE'S
021470                                   T4.1:
021470  104402                                   TRAP    C$BSUB

021472  012777  177400  160604          100$:    MOV     #177400,@A.FS           ;SET FUNCTION STATUS BITS F7->F0
021500  012737  177777  002734                   MOV     #-1,FP.EXP              ;INITIALIZE EXPECTED CONTENTS
021506  017737  160572  002736                   MOV     @A.FS,FP.ACT            ;READ REGISTER
021514  023737  002736  002734                   CMP     FP.ACT,FP.EXP           ;COMPARE WITH EXPECTED VALUE
021522  001424                                   BEQ     500$                    ;IDENTICAL:  CONTINUE

021524                                  300$:    ERRHRD  2,M.FSR1,ER.REG         ;"FUNCTION/STATUS BIT ERROR"
021524  104456                                   TRAP    C$ERHRD
021526  000002                                   .WORD   2
021530  010163                                   .WORD   M.FSR1
021532  015016                                   .WORD   ER.REG

021534  105737  002736                           TSTB    FP.ACT                  ;TEST STATUS BITS ONLY
021540  001015                                   BNE     500$                    ;AT LEAST ONE SET:  CONTINUE
021542                                   STARS
                                         ;;******************************************************************
                                         ;;
                                         ;;       ALL STATUS BITS LOW.  PROMPT OPERATOR TO CHECK
                                         ;;       THAT THE LOOP-BACK CABLE IS IN PLACE ON THE 8432 MODULE.
021542                                   STARS
                                         ;;******************************************************************

021542  005737  003050                           TST     CBLFLG                  ;TEST MESSAGE FLAG
021546  001012                                   BNE     500$                    ;SKIP PRINT IF NOT FIRST TIME

021550  005237  003050                           INC     CBLFLG                  ;BUMP FLAG
021554                                           PRINTF  #F.CBL                  ;"CHECK 8432 MODULE LOOP-BACK CABLE"
021554  012746  013464                           MOV     #F.CBL,-(SP)
021560  012746  000001                           MOV     #1,-(SP)
021564  010600                                   MOV     SP,R0
021566  104417                                   TRAP    C$PNTF
021570  062706  000004                           ADD     #4,SP

021574                                  500$:    ENDSUB                          ;ALL ONE'S
021574                                  L10032:
021574  104403                                   TRAP    C$ESUB
```

```
                                        STARS
                                        ;;**********************************************************************
                                        ;;
                                        ;;      SUBTEST 2:
                                        ;;
                                        ;;        LOAD FS REGISTER BITS F7-F0 WITH ZERO'S.
                                        ;;        READ REGISTER CONTENTS AND CHECK FOR
                                        ;;        FUNCTION/STATUS BITS STUCK AT ONE.
                                        STARS
                                        ;;**********************************************************************
                                                BGNSUB                          ;LOAD ALL ZERO'S
                                        T4.2:
                                                TRAP    C$BSUB

100$:           MOV     #0,@A.FS                ;CLEAR FUNCTION BITS F7->F0
                MOV     #0,FP.EXP               ;INITIALIZE EXPECTED VALUE
                MOV     @A.FS,FP.ACT            ;READ FS REGISTER
                CMP     FP.ACT,FP.EXP           ;AND COMPARE VALUES
                BEQ     400$                    ;IDENTICAL:  CONTINUE

                ERRHRD  3,M.FSR1,ER.REG         ;"FUNCTION/STATUS BIT ERROR"
                TRAP    C$ERHRD
                .WORD   3
                .WORD   M.FSR1
                .WORD   ER.REG

400$:           ENDSUB                          ;ALL ZERO'S
L10033:
                TRAP    C$ESUB
```

```
021644                            STARS
                                  ;;*************************************************************
                                  ;;
                                  ;;      SUBTEST 3:
                                  ;;
                                  ;;         LOAD FS REGISTER BITS F7-F0 WITH MOVING ONE.
                                  ;;         READ REGISTER CONTENTS AND CHECK FOR
                                  ;;         FUNCTION/STATUS BITS STUCK AT ZERO.
021644                            STARS
                                  ;;*************************************************************
021644                                   BGNSUB                       ;MOVING ONE BIT
021644                            T4.3:
021644  104402                           TRAP    C$BSUB
021646  005037  002470            50$:   CLR     R.ERR                ;CLEAR ERROR FLAG
021652  012702  000400                   MOV     #FS.F0,R2            ;INIT FUNCTION BIT F0
021656  012704  000401                   MOV     #401,R4              ;INIT EXPECTED BITS F0 & ST0
021662  012705  000010                   MOV     #8.,R5               ;AND BIT COUNTER
021666  010277  160412            100$:  MOV     R2,@A.FS             ;LOAD FS FUNCTION BIT
021672  010437  002734                   MOV     R4,FP.EXP            ;LOAD EXPECTED CONTENTS
021676  017737  160402  002736           MOV     @A.FS,FP.ACT         ;READ REGISTER
021704  023737  002736  002734           CMP     FP.ACT,FP.EXP        ;AND COMPARE VALUES
021712  001423                           BEQ     200$                 ;IDENTICAL:  CONTINUE
021714  005737  002470                   TST     R.ERR                ;TEST ERROR FLAG
021720  001016                           BNE     150$                 ;BRANCH IF NOT ZERO
021722  005237  002470                   INC     R.ERR                ;INCREMENT ERROR FLAG
021726                                   ERRHRD  4,M.FSR1             ;"FUNCTION/STATUS BIT ERROR"
021726  104456                           TRAP    C$ERHRD
021730  000004                           .WORD   4
021732  010163                           .WORD   M.FSR1
021734  000000                           .WORD   0
021736                                   PRINTX  #F.REG1              ;PRINT REGISTER HEADER
021736  012746  013056                   MOV     #F.REG1,-(SP)
021742  012746  000001                   MOV     #1,-(SP)
021746  010600                           MOV     SP,R0
021750  104415                           TRAP    C$PNTX
021752  062706  000004                   ADD     #4,SP
021756  004737  016054            150$:  JSR     PC,DSPREG            ;DISPLAY REGISTER CONTENTS
021762  006302                    200$:  ASL     R2                   ;SHIFT TO NEXT TEST PATTERN
021764  006304                           ASL     R4                   ;SHIFT TO NEXT EXPECTED POSITION
021766  005305                           DEC     R5                   ;DECREMENT COUNTER
021770  001336                           BNE     100$                 ;AND CONTINUE UNTIL ZERO
021772  005737  002470            400$:  TST     R.ERR                ;TEST FOR ANY ERRORS
021776  001410                           BEQ     600$                 ;NO:  EXIT TEST
022000                                   PRINTX  #F.CRLF              ;NEW LINE
022000  012746  014214                   MOV     #F.CRLF,-(SP)
022004  012746  000001                   MOV     #1,-(SP)
022010  010600                           MOV     SP,R0
022012  104415                           TRAP    C$PNTX
022014  062706  000004                   ADD     #4,SP
```

```
   3912 022020                          600$:   ENDSUB                          ;MOVING ONE
        022020                          L10034:
        022020  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 4:
                                        ;;
                                        ;;      LOAD FS REGISTER BITS F7-F0 WITH MOVING ZERO.
                                        ;;      READ REGISTER CONTENTS AND CHECK FOR
                                        ;;      FUNCTION/STATUS BITS STUCK AT ZERO.
                                        ;;
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;LOAD MOVING ZERO
                                        T4.4:
                                                TRAP    C$BSUB

                                        50$:    CLR     R.ERR                   ;CLEAR ERROR FLAG
                                                MOV     #177000,R2              ;INIT TEST BITS F7->F1
                                                MOV     #177376,R4              ;INIT EXPECTED BITS F7->F1 & ST7->ST1
                                                MOV     #8.,R5                  ;LOAD BIT COUNTER

                                        100$:   MOV     R2,@A.FS                ;WRITE TO FS REGISTER
                                                MOV     R4,FP.EXP               ;LOAD EXPECTED CONTENTS
                                                MOV     @A.FS,FP.ACT            ;READ REGISTER
                                                CMP     FP.ACT,FP.EXP           ;AND COMPARE VALUES
                                                BEQ     200$                    ;IDENTICAL:  CONTINUE

                                                TST     R.ERR                   ;TEST ERPOR FLAG
                                                BNE     150$                    ;BRANCH IF NOT ZERO

                                                INC     R.ERR                   ;INCREMENT ERROR FLAG
                                                ERRHRD  5,M.FSR1                ;"FUNCTION/STATUS BIT ERROR"
                                                TRAP    C$ERHRD
                                                .WORD   5
                                                .WORD   M.FSR1
                                                .WORD   0
                                                PRINTX  #F.REG1                 ;PRINT REGISTER HEADER
                                                MOV     #F.REG1,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
                                                ADD     #4,SP

                                        150$:   JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS

                                        200$:   ASL     R2                      ;SHIFT TO NEXT TEST POSITION
                                                BIS     #FS.F0,R2               ;SET INCOMING BIT F0
                                                ASL     R4                      ;SHIFT TO NEXT EXPECTED POSITION
                                                BIS     #FS.ST0,R4              ;SET INCOMING BIT ST0
                                                DEC     R5                      ;DECREMENT COUNTER
                                                BNE     100$                    ;AND CONTINUE UNTIL ZERO

                                        400$:   TST     R.ERR                   ;TEST FOR ANY ERRORS
                                                BEQ     600$                    ;NO:  EXIT TEST
                                                PRINTX  #F.CRLF                 ;NEW LINE
                                                MOV     #F.CRLF,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
```

```
                022202  062706  000004                  ADD     #4,SP
  3954
  3955          022206                          600$:   ENDSUB                          ;MOVING ZERO
                022206                          L10035:
                022206  104403                          TRAP    C$ESUB
  3956
  3957          022210  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
  3958
  3959          022214                                  ENDTST
                022214                          L10031:
                022214  104401                          TRAP    C$ETST
```

```
                                          .SBTTL  TEST  5:  CHECK INTERFACE STATUS ATTN BIT
     022216                               STARS
                                          ;;******************************************************************
                                          ;;
                                          ;;   TEST DESCRIPTION:
                                          ;;
                                          ;;       CHECK IS REGISTER USER ATTN BITS FOR BITS
                                          ;;       STUCK AT ONE, ZERO OR STUCK TOGETHER
                                          ;;
                                          ;;     TEST STEPS:
                                          ;;
                                          ;;       IF AT0 & AT3 ATTN NOT DISABLED THEN:
                                          ;;           CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                          ;;           EXECUTE SUB TESTS 1-4
                                          ;;
                                          ;;     ERROR CONDITIONS:
                                          ;;
                                          ;;       INTERFACE STATUS REGISTER BITS STUCK
     022216                               STARS
                                          ;;******************************************************************
     022216                                       BGNTST                              ;ATTN LOOP-BACK
     022216                               T5::

             000234                       SELMASK =       B.REVA + B.UPAR + B.BYTE + B.ABORT

     022216  013700  003010                       MOV     SELECT,R0                   ;READ SELECT WORD
     022222  042700  000234                       BIC     #SELMASK,R0                 ;CLEAR DON'T CARE SELECT BITS
     022226  001402                               BEQ     50$                         ;BRANCH IF B.AT0 AND B.AT3 NOT SET

     022230                                       EXIT    TST                         ;AT0 OR AT3 DISABLED:  EXIT TEST
     022230  104432                               TRAP    C$EXIT
     022232  000552                               .WORD   L10036-.

     022234  004737  015342           50$:        JSR     PC,CLRMBC                   ;CLEAR THE MASSBUS CONTROLLER
     022240  005737  003030                       TST     MBCFLG                      ;TEST CLEAR FLAG
     022244  001405                               BEQ     100$                        ;CONTINUE IF ZERO

     022246                                       ERRDF   1,M.MBC1,ER.DMP             ;"MASSBUS CONTROLLER CLEAR ERROR"
     022246  104455                               TRAP    C$ERDF
     022250  000001                               .WORD   1
     022252  011650                               .WORD   M.MBC1
     022254  015074                               .WORD   ER.DMP
     022256                                       DOCLN                               ;CLEAN UP AND EXIT
     022256  104444                               TRAP    C$DCLN

     022260  013737  002420  002730   100$:       MOV     NA.IS,FP.NAM                ;LOAD IS REGISTER NAME ADDRESS
     022266  013737  002450  002746               MOV     BA.IS,FP.TBL                ;LOAD BIT EXPAND TABLE
                                                                                      ;EXECUTE SUBTEST
```

```
                                        STARS
                                        ;;*****************************************************************
4001                                    ;;
4002                                    ;;      SUBTEST 1:
4003                                    ;;
4004                                    ;;        LOAD FS REGISTER FUNCTION BITS F3->F0 WITH ONE'S.
4005                                    ;;        READ AND CHECK IS REGISTER USER ATTN BITS AT3->AT0.
4006 022274                             STARS
                                        ;;*****************************************************************
4007 022274                                     BGNSUB                          ;LOAD ONE'S
     022274                             T5.1:
     022274  104402                             TRAP    C$BSUB
4008
4009 022276  012777  007400  160000    100$:   MOV     #7400,@A.FS             ;INITIALIZE FUNCTION BITS F3->F0
4010 022304  017737  160000  002736            MOV     @A.IS,FP.ACT            ;READ IS REGISTER
4011 022312  012737  142217  002734            MOV     #142217,FP.EXP          ;EXPECTED:  ATA,ERR,ERO,IRY & AT3->AT0
4012 022320  123737  002736  002734            CMPB    FP.ACT,FP.EXP           ;AND COMPARE LSB VALUES
4013 022326  001404                             BEQ     200$                    ;IDENTICAL:  CONTINUE
4014
4015 022330                                     ERRHRD  2,M.UAT1,ER.REG         ;"ISR USER ATTN BIT NOT SET/RESET"
     022330  104456                             TRAP    C$ERHRD
     022332  000002                             .WORD   2
     022334  012454                             .WORD   M.UAT1
     022336  015016                             .WORD   ER.REG
4016
4017 022340                             200$:   ENDSUB                          ;ALL ONE'S
     022340                             L10037:
     022340  104403                             TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 2:
                                        ;;
                                        ;;        LOAD FS REGISTER FUNCTION BITS F3->F0   WITH ZERO'S.
                                        ;;        READ AND CHECK IS REGISTER USER ATTN BITS AT3->AT0.
                                        STARS
                                        ;;*****************************************************************
                                T5.2:   BGNSUB                          ;ALL ZERO'S
                                        TRAP    C$BSUB
                                100$:   MOV     #0,@A.FS                ;INITIALIZE BITS F7->F0 TO ZERO
                                        MOV     @A.IS,FP.ACT            ;READ IS REGISTER
                                        MOV     #142200,FP.EXP          ;EXPECTED:  ATA,ERR,ERO & IRY
                                        CMPB    FP.ACT,FP.EXP           ;AND COMPARE LSB VALUES
                                        BEQ     200$                    ;IDENTICAL:  CONTINUE
                                        ERRHRD  3,M.UAT1,ER.REG         ;"ISR USER ATTN BIT NOT SET/RESET"
                                        TRAP    C$ERHRD
                                        .WORD   3
                                        .WORD   M.UAT1
                                        .WORD   ER.REG
                                200$:   ENDSUB                          ;ALL ZERO'S
                                L10040:
                                        TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 3:
                                        ;;
                                        ;;      LOAD FS REGISTER BITS F3->F0 WITH MOVING ONE BIT.
                                        ;;      READ AND CHECK IS REGISTER USER ATTN BITS AT3->AT0.
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;MOVING ONE BIT
                                        T5.3:
                                                TRAP    C$BSUB

                                        50$:    CLR     R.ERR                   ;CLEAR ERROR FLAG
                                                MOV     #FS.F0,R1               ;INITIALIZE BIT F0 TO ONE
                                                MOV     #IS.AT0,R2              ;AND EXPECTED ATTN BIT AT0 TO ONE
                                                MOV     #4,R5                   ;INITIALIZE BIT COUNTER

                                        100$:   MOV     R1,@A.FS                ;WRITE TO FS REGISTER
                                                MOV     @A.IS,FP.ACT            ;READ IS REGISTER
                                                MOV     #142200,FP.EXP          ;EXPECTED:  ATA,ERR,ERO & IRY
                                                BIS     R2,FP.EXP               ;AND SET EXPECTED ATTN BIT
                                                CMPB    FP.ACT,FP.EXP           ;AND COMPARE LSB VALUES
                                                BEQ     200$                    ;IDENTICAL:  CONTINUE

                                                TST     R.ERR                   ;TEST ERROR FLAG
                                                BNE     150$                    ;BRANCH IF NOT ZERO

                                                INC     R.ERR                   ;INCREMENT ERROR FLAG
                                                ERRHRD  4,M.UAT1                ;"ISR USER ATTN BIT NOT SET/RESET"
                                                TRAP    C$ERHRD
                                                .WORD   4
                                                .WORD   M.UAT1
                                                .WORD   0
                                                PRINTX  #F.REG1                 ;PRINT REGISTER HEADER
                                                MOV     #F.REG1,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
                                                ADD     #4,SP

                                        150$:   JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS

                                        200$:   ASL     R1                      ;SHIFT TO NEXT FUNCTION BIT
                                                ASL     R2                      ;SHIFT TO NEXT EXPECTED ATTN BIT
                                                DEC     R5                      ;DECREMENT COUNTER
                                                BNE     100$                    ;AND CONTINUE UNTIL ZERO

                                        400$:   TST     R.ERR                   ;TEST FOR ANY ERRORS
                                                BEQ     600$                    ;NO:  EXIT TEST
                                                PRINTX  #F.CRLF                 ;NEW LINE
                                                MOV     #F.CRLF,-(SP)
                                                MOV     #1,-(SP)
                                                MOV     SP,R0
                                                TRAP    C$PNTX
                                                ADD     #4,SP
```

```
 4077 022572                          600$:   ENDSUB                          ;MOVING ONE BIT
      022572                          L10041:
      022572  104403                         TRAP    C$ESUB
```

```
4079 022574                                STARS
                                           ;;*****************************************************************
4080                                       ;;
4081                                       ;;    SUBTEST 4:
4082                                       ;;
4083                                       ;;      LOAD FS REGISTER BITS F3->F0 WITH MOVING ZERO BIT.
4084                                       ;;      READ AND CHECK IS REGISTER USER ATTN BITS AT3->AT0.
4085 022574                                STARS
                                           ;;*****************************************************************
4086 022574                                        BGNSUB                          ;MOVING ZERO BIT
     022574                                T5.4:
     022574  104402                             TRAP    C$BSUB
4087
4088 022576  005037  002470          50$:    CLR     R.ERR                   ;CLEAR ERROR FLAG
4089 022602  012701  007000                  MOV     #7000,R1                ;INITIALIZE BITS F3-F1 TO ONE
4090 022606  012702  000016                  MOV     #16,R2                  ;INITIALIZE EXPECTED ATTN BITS AT3->AT1
4091 022612  012705  000004                  MOV     #4,R5                   ;AND BIT COUNTER
4092
4093 022616  010177  157462          100$:   MOV     R1,@A.FS                ;WRITE TO FS REGISTER
4094 022622  017737  157462  002736          MOV     @A.IS,FP.ACT            ;READ IS REGISTER
4095 022630  012737  142200  002734          MOV     #142200,FP.EXP          ;EXPECTED:  ATA,ERR,ERO & IRY
4096 022636  060237  002734                  ADD     R2,FP.EXP               ;SET EXPECTED AT3->AT0
4097 022642  123737  002736  002734          CMPB    FP.ACT,FP.EXP           ;AND COMPARE LSB VALUES
4098 022650  001423                          BEQ     200$                    ;IDENTICAL:  CONTINUE
4099
4100 022652  005737  002470                  TST     R.ERR                   ;TEST ERROR FLAG
4101 022656  001016                          BNE     150$                    ;BRANCH IF NOT ZERO
4102
4103 022660  005237  002470                  INC     R.ERR                   ;INCREMENT ERROR FLAG
4104 022664                                  ERRHRD  5,M.UAT1                ;"ISR USER ATTN BIT NOT SET/RESET"
     022664  104456                          TRAP    C$ERHRD
     022666  000005                          .WORD   5
     022670  012454                          .WORD   M.UAT1
     022672  000000                          .WORD   0
4105 022674                                  PRINTX  #F.REG1                 ;PRINT REGISTER HEADER
     022674  012746  013056                  MOV     #F.REG1,-(SP)
     022700  012746  000001                  MOV     #1,-(SP)
     022704  010600                          MOV     SP,R0
     022706  104415                          TRAP    C$PNTX
     022710  062706  000004                  ADD     #4,SP
4106
4107 022714  004737  016054          150$:   JSR     PC,DSPREG               ;DISPLAY REGISTER CONTENTS
4108
4109 022720  006301                  200$:   ASL     R1                      ;SHIFT TO NEXT FUNCTION BIT
4110 022722  042701  010000                  BIC     #FS.F4,R1               ;CLEAR SHIFT BIT
4111 022726  052701  000400                  BIS     #FS.F0,R1               ;AND SET INCOMING FUNCTION BIT
4112 022732  006302                          ASL     R2                      ;SHIFT TO NEXT ATTN BIT
4113 022734  042702  000020                  BIC     #IS.CYC,R2              ;CLEAR SHIFT BIT
4114 022740  052702  000001                  BIS     #IS.AT0,R2              ;AND SET INCOMING ATTN BIT
4115 022744  005305                          DEC     R5                      ;DECREMENT COUNTER
4116 022746  001323                          BNE     100$                    ;AND CONTINUE UNTIL ZERO
4117
4118 022750  005737  002470          400$:   TST     R.ERR                   ;TEST FOR ANY ERRORS
4119 022754  001410                          BEQ     600$                    ;NO:  EXIT TEST
4120 022756                                  PRINTX  #F.CRLF                 ;NEW LINE
     022756  012746  014214                  MOV     #F.CRLF,-(SP)
     022762  012746  000001                  MOV     #1,-(SP)
```

TEST 5: CHECK INTERFACE STATUS ATTN BIT

```
        022766  010600                          MOV     SP,R0
        022770  104415                          TRAP    C$PNTX
        022772  062706  000004                  ADD     #4,SP
  4121
  4122  022776                        600$:     ENDSUB                          ;MOVING ZERO BIT
        022776                        L10042:
        022776  104403                          TRAP    C$ESUB
  4123
  4124  023000  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
  4125
  4126  023004                                  ENDTST
        023004                        L10036:
        023004  104401                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST  6:  CHECK INTERFACE STATUS USER ATTN BITS
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  TEST DESCRIPTION:
                                        ;;
                                        ;;      CHECK THAT USER ATTN BITS AT3->AT0 SET ATA
                                        ;;      CHECK THAT MASSBUS CONTROLLER CLEAR RESETS ATA
                                        ;;
                                        ;;    TEST STEPS:
                                        ;;
                                        ;;      IF AT0 & AT3 NOT DISABLED THEN:
                                        ;;          FOR EACH USER ATTN BIT:  EXECUTE SUBTEST 1-2
                                        ;;
                                        ;;    ERROR CONDITIONS:
                                        ;;
                                        ;;      USER ATTN BIT DOES NOT SET ATA
                                        ;;      INTERFACE CLEAR COMMAND DOES NOT RESET ATA
                                        STARS
                                        ;;*****************************************************************
                                                BGNTST
023006                                  T6::

                000234                  SELMASK =       B.REVA + B.UPAR + B.BYTE + B.ABORT

023006  013700  003010                          MOV     SELECT,R0               ;READ SELECT WORD
023012  042700  000234                          BIC     #SELMASK,R0             ;CLEAR DON'T CARE SELECT BITS
023016  001402                                  BEQ     50$                     ;BRANCH IF NOT SET

023020                                          EXIT    TST                     ;AT0 OR AT3 DISABLED:  EXIT TEST
023020  104432                                  TRAP    C$EXIT
023022  000302                                  .WORD   L10043-.

023024  012701  000400          50$:    MOV     #FS.F0,R1               ;INIT FS REGISTER TEST BIT F0
023030  012702  000001                  MOV     #IS.AT0,R2              ;INIT IS EXPECTED BIT AT0
023034  012705  000004                  MOV     #4,R5                   ;LOAD BIT COUNTER

023040  013737  002420  002730  100$:   MOV     NA.IS,FP.NAM            ;LOAD IS REGISTER NAME POINTER
023046  013737  002450  002746          MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST

023054                          LP.UAT:                                 ;USER ATTN LOOP LABEL
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      INITIALIZE THE MASSBUS CONTROLLER AND CHECK FOR ERROR STATUS
                                        ;;      SET ONE OF F3->F0 IN THE FS REGISTER
                                        ;;      TEST CORRESPONDING USER ATTN BIT AT3->AT0 SET
                                        ;;      CHECK THAT USER ATTN BIT SETS ATA
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;SET ATTENTION
                                        T6.1:
023054  104402                                 TRAP    C$BSUB

023056  004737  015342                 100$:   JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
023062  005737  003030                         TST     MBCFLG                  ;TEST FOR CONTROLLER ERROR
023066  001405                                 BEQ     20U$                    ;CONTINUE IF MASSBUS CONTROLLER CLEARED

023070                                          ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
023070  104455                                 TRAP    C$ERDF
023072  000001                                 .WORD   1
023074  011650                                 .WORD   M.MBC1
023076  015074                                 .WORD   ER.DMP
023100                                          DOCLN                           ;CLEAN UP AND EXIT
023100  104444                                 TRAP    C$DCLN

023102  010177  157176          20C$:   MOV     R1,@A.FS                ;WRITE USER FUNCTION BIT IN FS REGISTER
023106  017737  157176  002736          MOV     @A.IS,FP.ACT            ;READ IS REGISTER
023114  012737  100200  002734          MOV     #100200,FP.EXP          ;EXPECTED:  ATA & IRY
023122  050237  002734                  BIS     R2,FP.EXP               ;SET CORRESPONDING USET ATTN BIT
023126  123737  002736  002734          CMPB    FP.ACT,FP.EXP           ;COMPARE LSB VALUES ONLY
023134  001405                          BEQ     300$                    ;IDENTICAL:  CONTINUE

023136                                  ERRHRD  2,M.UAT1,ER.REG         ;''IS REGISTER USER ATTN BIT NOT SET''
023136  104456                          TRAP    C$ERHRD
023140  000002                          .WORD   2
023142  012454                          .WORD   M.UAT1
023144  015016                          .WORD   ER.REG
023146  000410                          BR      400$                    ;EXIT SUBTEST

023150  023737  002736  002734  300$:   CMP     FP.ACT,FP.EXP           ;USER ATTN OK:  TEST ATA
023156  001404                          BEQ     400$                    ;ATA OK IF SET

023160                                  ERRHRD  3,M.ISR1,ER.REG         ;''INTERFACE STATUS ERROR''
023160  104456                          TRAP    C$ERHRD
023162  000003                          .WORD   3
023164  011543                          .WORD   M.ISR1
023166  015016                          .WORD   ER.REG

023170                          400$:   ENDSUB                          ;SET ATTENTION
023170                          L10044:
023170  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      IF NOT REVA THEN:  ISSUE INTERFACE CLEAR CMD AND CHECK FOR ERROR
                                        ;;        ELSE:  CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      READ IS REGISTERS AND CHECK THAT ATA BIT RESET
                                        STARS
                                        ;;*****************************************************************

                                        500$:   BGNSUB                          ;RESET ATTENTION
                                        T6.2:
                                                TRAP    C$BSUB

                                                TST     REVFLG                  ;TEST FOR REVA
                                                BNE     550$                    ;BRANCH IF REVA

                                        ;       REVISION 'B'                    ;ISSUE CLEAR DRIVE AND CONTINUE

                                        525$:   JSR     PC,CLRDRI               ;ISSUE DRIVE CLEAR AND TEST ERROR
                                                TST     DRIFLG                  ;INTERFACE CLEAR ERROR
                                                BEQ     600$                    ;CONTINUE IF NO ERROR

                                                ERRHRD  4,M.ISR2,ER.DMP         ;''INTERFACE STATUS ERROR ON CLEAR CMD''
                                                TRAP    C$ERHRD
                                                .WORD   4
                                                .WORD   M.ISR2
                                                .WORD   ER.DMP
                                                BR      700$                    ;EXIT TEST

                                        ;       REVISION ''A''                  ;CLEAR MASSBUS CONTROLLER AND CONTINUE

                                        550$:   JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
                                                TST     MBCFLG                  ;TEST FOR CONTROLLER ERROR
                                                BEQ     600$                    ;CONTINUE IF MASSBUS CONTROLLER CLEARED

                                                ERRDF   5,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
                                                TRAP    C$ERDF
                                                .WORD   5
                                                .WORD   M.MBC1
                                                .WORD   ER.DMP
                                                DOCLN                           ;CLEAN UP AND EXIT
                                                TRAP    C$DCLN

                                        600$:   MOV     @A.IS,FP.ACT            ;READ IS REGISTER
                                                MOV     #IS.IRY,FP.EXP          ;LOAD EXPECTED CONTENTS
                                                CMP     FP.ACT,FP.EXP           ;COMPARE VALUES
                                                BEQ     700$                    ;CONTINUE IF ATA RESET

                                                ERRHRD  6,M.ATA3,ER.REG         ;''ATA NOT RESET BY MBC CLEAR CMD''
                                                TRAP    C$ERHRD
                                                .WORD   6
                                                .WORD   M.ATA3
                                                .WORD   ER.REG

                                        700$:   ENDSUB                          ;RESET ATTENTION
```

```
                023306                          L10045:
                023306  104403                          TRAP    C$ESUB
          4241
          4242  023310  006301                          ASL     R1              ;SHIFT TO NEXT TEST POSITION
          4243  023312  006302                          ASL     R2              ;SHIFT TO NEXT EXPECTED POSITION
          4244  023314  005305                          DEC     R5              ;DECREMENT COUNTER
          4245  023316  001256                          BNE     LP.UAT          ;CONTINUE LOOP UNTIL ZERO
          4246
          4247  023320  004737  015550          800$:   JSR     PC,CLRTST       ;AND CLEAR TEST DATA
          4248
          4249  023324                          ENDTST
                023324                          L10043:
                023324  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  7:  CHECK INTERFACE STATUS ERR BIT
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK PROPER FUNCTION OF ERR BIT
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      CLEAR MASSBUS CONTROLLER AND TEST FOR ERROR
                                ;;      EXECUTE SUBTEST 1-4
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      INTERFACE STATUS ERROR ON DRIVE CLEAR
                                ;;      ERO NOT SET THROUGH M8432 /NOT RESET BY INTERFACE CLEAR
                                ;;      ERR NOT BY ERO /NOT RESET BY INTERFACE CLEAR CMD
                                ;;      ATA NOT SET BY ERR /NOT RESET BY INTERFACE CLEAR CMD
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T7::
                                50$:    JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
                                        TST     MBCFLG                  ;TEST FOR CONTROLLER ERROR
                                        BEQ     100$                    ;CONTINUE IF MASSBUS CONTROLLER CLEARED

                                        ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
                                        TRAP    C$ERDF
                                        .WORD   1
                                        .WORD   M.MBC1
                                        .WORD   ER.DMP
                                        DOCLN                           ;CLEAN UP AND EXIT
                                        TRAP    C$DCLN

                                100$:   MOV     NA.IS,FP.NAM            ;LOAD IS REGISTER NAME POINTER
                                        MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      SET BIT F5 OF FS REGISTER
                                        ;;      CHECK IS REGISTER ERO BIT SET (8432 LOOP-BACK)
                                        ;;      CHECK IS REGISTER ERR BIT SET (ERO)
                                        ;;      CHECK IS REGISTER ATA BIT SET (ERR)
                                        STARS
                                        ;;*****************************************************************
                                        T7.1:   BGNSUB                          ;SET ERROR
                                                TRAP    C$BSUB
                                100$:   MOV     #FS.F5,@A.FS                    ;SET FS REGISTER BIT F5
                                        MOV     @A.IS,FP.ACT                    ;READ IS REGISTER
                                        MOV     #142200,FP.EXP                  ;LOAD EXPECTED CONTENTS
                                        MOV     #0,@A.FS                        ;CLEAR FUNCTION / STATUS BITS
                                200$:   BIT     #IS.ERO,FP.ACT                  ;TEST FOR ERO BIT SET
                                        BNE     250$                            ;YES:  CONTINUE
                                        ERRHRD  2,M.ERO1,ER.REG                 ;"ERO NOT SET WHEN FS BIT13 SET"
                                        TRAP    C$ERHRD
                                        .WORD   2
                                        .WORD   M.ERO1
                                        .WORD   ER.REG
                                        BR      400$                            ;CONTINUE
                                250$:   BIT     #IS.ERR,FP.ACT                  ;TEST FOR ERR BIT SET
                                        BNE     350$                            ;YES:  CONTINUE
                                        ERRHRD  3,M.ERR1,ER.REG                 ;"ERR NOT SET ON ERO ACTIVE"
                                        TRAP    C$ERHRD
                                        .WORD   3
                                        .WORD   M.ERR1
                                        .WORD   ER.REG
                                        BR      400$                            ;CONTINUE
                                350$:   BIT     #IS.ATA,FP.ACT                  ;TEST ATA BIT SET
                                        BNE     400$                            ;YES:  WRAP IT UP
                                        ERRHRD  4,M.ATA2,ER.REG                 ;"ATA NOT SET ON ERR ACTIVE"
                                        TRAP    C$ERHRD
                                        .WORD   4
                                        .WORD   M.ATA2
                                        .WORD   ER.REG
                                400$:   ENDSUB                                  ;SET ERROR
                                L10047:
                                        TRAP    C$ESUB
```

```
 4319 023506                            STARS
                                        ;;**************************************************************
 4320                                   ;;
 4321                                   ;;    SUBTEST 2:
 4322                                   ;;
 4323                                   ;;      IF NOT REVA THEN:  ISSUE INTERFACE CLEAR COMMAND
 4324                                   ;;          ELSE:  CLEAR MASSBUS CONTROLLER
 4325                                   ;;
 4326                                   ;;      CHECK IS REGISTER ERO BIT RESET
 4327                                   ;;      CHECK IS REGISTER ERR BIT RESET
 4328                                   ;;      CHECK IS REGISTER ATA BIT RESET
 4329 023506                            STARS
                                        ;;**************************************************************
 4330 023506                                    BGNSUB                          ;INTERFACE CLEAR ERROR
      023506                            T7.2:
      023506  104402                            TRAP    C$BSUB
 4331
 4332 023510  005737  002226            50$:    TST     REVFLG                  ;TEST FOR REVA
 4333 023514  001012                            BNE     80$                     ;BRANCH IF REVA
 4334
 4335 023516  004737  015464                    JSR     PC,CLRDRI               ;ISSUE DRIVE CLEAR AND CHECK ERROR
 4336 023522  005737  003032                    TST     DRIFLG                  ;TEST ERROR FLAG
 4337 023526  001417                            BEQ     100$                    ;CONTINUE IF NO ERROR
 4338
 4339 023530                                    ERRHRD  5,M.ISR2,ER.DMP         ;''INTERFACE STATUS ERROR ON CLEAR''
      023530  104456                            TRAP    C$ERHRD
      023532  000005                            .WORD   5
      023534  011572                            .WORD   M.ISR2
      023536  015074                            .WORD   ER.DMP
 4340 023540  000452                            BR      400$                    ;EXIT TEST ON ERROR
 4341
 4342 023542  004737  015342            80$:    JSR     PC,CLRMBC               ;CLEAR THE MASSBUS CONTROLLER
 4343 023546  005737  003030                    TST     MBCFLG                  ;TEST FOR CONTROLLER ERROR
 4344 023552  001405                            BEQ     100$                    ;CONTINUE IF MASSBUS CONTROLLER CLEARED
 4345
 4346 023554                                    ERRDF   6,M.MBC1,ER.DMP         ;''MASSBUS CONTROLLER CLEAR ERROR''
      023554  104455                            TRAP    C$ERDF
      023556  000006                            .WORD   6
      023560  011650                            .WORD   M.MBC1
      023562  015074                            .WORD   ER.DMP
 4347 023564                                    DOCLN                           ;CLEAN UP AND EXIT
      023564  104444                            TRAP    C$DCLN
 4348
 4349 023566  017737  156516  002736  100$:    MOV     @A.IS,FP.ACT            ;READ IS REGISTER
 4350 023574  012737  000200  002734            MOV     #IS.IRY,FP.EXP          ;LOAD EXPECTED CONTENTS
 4351
 4352 023602  032737  002000  002736  200$:    BIT     #IS.ERO,FP.ACT          ;TEST ERO BIT RESET
 4353 023610  001405                            BEQ     250$                    ;BRANCH IF ERO RESET
 4354
 4355 023612                                    ERRHRD  7,M.ERO2,ER.REG         ;''ERO NOT RESET ON INTERFACE CLEAR CMD''
      023612  104456                            TRAP    C$ERHRD
      023614  000007                            .WORD   7
      023616  010017                            .WORD   M.ERO2
      023620  015016                            .WORD   ER.REG
 4356 023622  000421                            BR      400$                    ;CONTINUE
 4357
 4358 023624  032737  040000  002736  250$:    BIT     #IS.ERR,FP.ACT          ;TEST FOR ERR BIT RESET
```

```
4359 023632  001405                          BEQ     300$                    ;BRANCH IF ERR RESET
4360
4361 023634                                  ERRHRD  8,M.ERR2,ER.REG         ;''ERR NOT RESET ON INTERFACE CLEAR CMD''
     023634  104456                          TRAP    C$ERHRD
     023636  000010                          .WORD   8
     023640  010116                          .WORD   M.ERR2
     023642  015016                          .WORD   ER.REG
4362 023644  000410                          BR      400$                    ;CONTINUE
4363
4364 023646  032737  100000  002736  300$:   BIT     #IS.ATA,FP.ACT          ;TEST FOR ATA BIT RESET
4365 023654  001404                          BEQ     400$                    ;BRANCH IF RESET
4366
4367 023656                                  ERRHRD  9,M.ATA3,ER.REG         ;''ATA NOT RESET ON INTERFACE CLEAR CMD''
     023656  104456                          TRAP    C$ERHRD
     023660  000011                          .WORD   9
     023662  006740                          .WORD   M.ATA3
     023664  015016                          .WORD   ER.REG
4368
4369 023666                          400$:   ENDSUB                          ;INTERFACE CLEAR ERROR
     023666                          L10050:
     023666  104403                          TRAP    C$ESUB
4370
4371 023670  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
4372
4373 023674                                  ENDTST
     023674                          L10046:
     023674  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  8:  CHECK INTERFACE STATUS IRY BIT ON ERROR
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK PROPER FUNCTION OF IRY BIT
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                ;;      EXECUTE SUBTEST 1-2
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      IRY NOT RESET BY VALID DATA TRANSFER CMD
                                ;;      IRY NOT SET BY ERR STATUS
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T8::
                                50$:    JSR     PC,CLRMBC               ;AND CLEAR MASSBUS CONTROLLER
                                        TST     MBCFLG                  ;TEST FOR ERROR
                                        BEQ     100$                    ;BRANCH IF ZERO
                                        ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
                                        TRAP    C$ERDF
                                        .WORD   1
                                        .WORD   M.MBC1
                                        .WORD   ER.DMP
                                        DOCLN                           ;CLEAN UP AND EXIT
                                        TRAP    C$DCLN
                                100$:   MOV     NA.IS,FP.NAM            ;LOAD IS REGISTER NAME POINTER
                                        MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST
```

```
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;    SUBTEST 1:
                                ;;
                                ;;      ISSUE WRITE COMMAND TO INTERFACE (NO DMA)
                                ;;      READ IS REGISTER AND CHECK IRY RESET
                                ;;
                                STARS
                                ;;*****************************************************************
                                        BGNSUB                          ;IRY RESET
                                T8.1:
023736  104402                          TRAP    C$BSUB

023740  012777  000061  156330          MOV     #WR.CMD,@A.CS1          ;ISSUE WRITE CMD TO INTERFACE
023746  004737  016042                  JSR     PC,DLY                  ;DELAY A COUPLE OF MEMORY CYCLES

023752  004737  015606                  JSR     PC,RDREG                ;READ REGISTERS
023756  032737  000200  002340          BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT RESET
023764  001404                          BEQ     200$                    ;RESET:  CONTINUE

023766                                  ERRHRD  2,M.IRY3,ER.DMP         ;"IRY NOT RESET BY DATA TRANSFER CMD"
023766  104456                          TRAP    C$ERHRD
023770  000002                          .WORD   2
023772  011162                          .WORD   M.IRY3
023774  015074                          .WORD   ER.DMP

023776                          200$:   ENDSUB                          ;IRY FESET
023776                          L10052:
023776  104403                          TRAP    C$ESUB
```

```
4426 024000                          STARS
                                     ;;*****************************************************************
4427                                 ;;
4428                                 ;;    SUBTEST 2:
4429                                 ;;
4430                                 ;;      SET ERO THROUGH LOOP-BACK CABLE
4431                                 ;;      READ IS REGISTER AND CHECK IRY SET
4432 024000                          STARS
                                     ;;*****************************************************************
4433 024000                                  BGNSUB                              ;IRY SET
     024000                          T8.2:
     024000  104402                          TRAP    C$BSUB
4434
4435 024002  012777  020000  156274  100$:   MOV     #FS.F5,@A.FS                ;SET FUNCTION BIT 5 IN FS REGISTER
4436 024010  004737  015606                  JSR     PC,RDREG                    ;READ REGISTERS
4437 024014  012777  000000  156262          MOV     #0,@A.FS                    ;CLEAR FUNCTION / STATUS BITS
4438
4439 024022  032737  000200  002340          BIT     #IS.IRY,R.IS                ;AND TEST FOR IRY SET
4440 024030  001004                          BNE     200$                        ;CONTINUE IF SET
4441
4442 024032                                  ERRHRD  3,M.IRY1,ER.DMP             ;"IRY NOT SET ON ERROR"
     024032  104456                          TRAP    C$ERHRD
     024034  000003                          .WORD   3
     024036  011034                          .WORD   M.IRY1
     024040  015074                          .WORD   ER.DMP
4443
4444 024042                          200$:   ENDSUB
     024042                          L10053:
     024042  104403                          TRAP    C$ESUB
4445
4446 024044  004737  015550                  JSR     PC,CLRTST                   ;AND CLEAR TEST DATA
4447
4448 024050                                  ENDTST
     024050                          L10051:
     024050  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST  9:  ADVANCED ATA BIT FUNCTIONAL CHECKS
024052                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      VERIFY ATA BIT WILL CAUSE INTERRUPT WHEN INTERRUPTS ENABLED,
                                ;;      THAT ATA BIT IS SET/ RESET WITH ATTENTION SUMMARY REGISTER,
                                ;;      AND THAT ATA BIT CLEARS WITH VALID DATA TRANSFER COMMAND.
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      EXECUTE SUBTEST 1-4
                                ;;
                                ;;    ERROR CONDITIONS·
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      NO INTERRUPT ON ATA SET
                                ;;      AS REGISTER ATTN SUMMARY BIT NOT SET
                                ;;      AS REGISTER ATA NOT SET/RESET BY AS BIT
024052                          STARS
                                ;;*****************************************************************
024052                                  BGNTST
024052                          T9::

024052                          100$:                                   ;EXECUTE SUBTEST
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      SETUP INTERRUPT SERVICE VECTOR
                                        ;;      SET FS REGISTER BIT F5 TO SET ERO, ERR & ATA
                                        ;;      DELAY AND READ REGISTERS
                                        ;;      TEST FOR INTERRUPT AND CLEAR SERVICE VECTOR
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;CHECK INTERRUPT
                                        T9.1:
                                                TRAP    C$BSUB
                                        50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
                                                TST     MBCFLG                  ;TEST FOR ERROR
                                                BEQ     100$                    ;CONTROLLER CLEAR:  CONTINUE

                                                ERRDF   1,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
                                                TRAP    C$ERDF
                                                .WORD   1
                                                .WORD   M.MBC1
                                                .WORD   ER.DMP
                                                DOCLN                           ;CLEAN UP AND EXIT
                                                TRAP    C$DCLN

                                        100$:   CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
                                                JSR     PC,SETISR               ;SET INTERRUPT SERVICE VECTOR

                                                MOV     #CS1.IE,@A.CS1          ;WRITE INTERRUPT ENABLE TO CS1
                                                MOV     #FS.F5,@A.FS            ;SET FUNCTION BIT F5 IN FS REGISTER
                                                JSR     PC,DLY                  ;DELAY A FEW CYCLES
                                                JSR     PC,RDREG                ;AND READ REGISTERS

                                                MOV     #0,@A.FS                ;CLEAR FUNCTION / STATUS BITS

                                                TST     INTFLG                  ;TEST INTERRUPT FLAG
                                                BNE     200$                    ;CONTINUE IF SET

                                                ERRHRD  2,M.ATA4,ER.DMP         ;'NO INTERRUPT ON ATA''
                                                TRAP    C$ERHRD
                                                .WORD   2
                                                .WORD   M.ATA4
                                                .WORD   ER.DMP

                                        200$:   JSR     PC,CLRISR               ;CLEAR INTERRUPT SERVICE VECTOR

                                                ENDSUB                          ;CHECK INTERRUPTS
                                        L10055:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;**************************************************************************
                                        ;;
                                        ;;      SUBTEST 2:
                                        ;;
                                        ;;        READ AS REGISTER CONTENTS
                                        ;;        CHECK CORRESPONDING AS REGISTER BIT SET
                                        STARS
                                        ;;**************************************************************************
                                                BGNSUB                          ;ATTENTION SUMMARY BITS
                                        T9.2:
                                                TRAP    C$BSUB

                                        100$:   MOV     UNITPOS,FP.EXP          ;LOAD EXPECTED AS REGISTER BIT
                                                MOV     @A.AS,FP.ACT            ;READ AS REGISTER CONTENTS
                                                CMP     FP.ACT,FP.EXP           ;AND COMPARE VALUES
                                                BEQ     200$                    ;IDENTICAL:  CONTINUE

                                                MOV     NA.AS,FP.NAM            ;LOAD REGISTER NAME ADDRESS
                                                MOV     BA.AS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                ERRHRD  3,M.ASB1,ER.REG         ;"AS BIT NOT SET"
                                                TRAP    C$ERHRD
                                                .WORD   3
                                                .WORD   M.ASB1
                                                .WORD   ER.REG

                                        200$:   ENDSUB                          ;AS BIT SET
                                        L10056:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 3:
                                        ;;
                                        ;;        WRITE TO ATTENTION SUMMARY BIT IN AS REGISTER
                                        ;;        READ IS REGISTER CONTENTS, CHECK FOR ATA BIT RESET
                                        ;;        AND CHECK FOR ERR BIT SET
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;ATA BIT RESET
                                        T9.3:
                                                TRAP    C$BSUB

                                        100$:   MOV     UNITPOS,@A.AS           ;WRITE TO ATTN SUMMARY BIT
                                                MOV     @A.IS,FP.ACT            ;READ IS REGISTER CONTENTS
                                                MOV     #42200,FP.EXP           ;EXPECTED VALUE:  (ERR,ERO & IRY)
                                                TST     REVFLG                  ;TEST FOR REV "A"
                                                BEQ     150$                    ;CONTINUE IF NOT
                                                MOV     #40200,FP.EXP           ;RESET ERO IN EXPECTED VALUE

                                        150$:   BIT     #IS.ATA,FP.ACT          ;TEST FOR ATA RESET
                                                BEQ     200$                    ;CONTINUE IF ZERO

                                                ERRHRD  4,M.ATA5                ;"ATA NOT RESET BY WRITING AS BIT"
                                                TRAP    C$ERHRD
                                                .WORD   4
                                                .WORD   M.ATA5
                                                .WORD   0
                                                BR      250$                    ;DISPLAY INTERFACE STATUS

                                        200$:   CMP     FP.ACT,FP.EXP           ;COMPARE VALUES
                                                BEQ     400$                    ;BRANCH IF IDENTICAL

                                        250$:   MOV     NA.IS,FP.NAM            ;LOAD REGISTER NAME ADDRESS
                                                MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                ERRHRD  5,M.ISR1,ER.REG         ;"INTERFACE STATUS ERROR"
                                                TRAP    C$ERHRD
                                                .WORD   5
                                                .WORD   M.ISR1
                                                .WORD   ER.REG

                                        400$:   ENDSUB                          ;ATA BIT RESET
                                        L10057:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 4:
                                        ;;
                                        ;;
                                        ;;        CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR STATUS
                                        ;;        WRITE ATO BIT TO SET ATA AND ISSUE WRITE COMMAND TO CS1
                                        ;;        READ IS REGISTER CONTENTS AND CHECK FOR ATA RESET
                                        ;;
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;ATA RESET/VALID XFER COMMAND
                                        T9.4:
                                                TRAP    C$BSUB

024364  004737  015342          50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
024370  005737  003030                  TST     MBCFLG                  ;TEST FOR ERROR
024374  001405                          BEQ     100$                    ;CONTROLLER CLEAR:  CONTINUE

024376                                  ERRDF   6,M.MBC1,ER.DMP         ;'MASSBUS CONTROLLER CLEAR ERROR''
024376  104455                          TRAP    C$ERDF
024400  000006                          .WORD   6
024402  011650                          .WORD   M.MBC1
024404  015074                          .WORD   ER.DMP
024406                                  DOCLN                           ;CLEAN UP AND EXIT
024406  104444                          TRAP    C$DCLN

024410  012777  000001  155672  100$:   MOV     #IS.ATO,@A IS           ;SET ATA BIT IN IS REGISTER

024416  012777  000061  155652          MOV     #WR.CMD,@A.CS1          ;ISSUE WRITE COMMAND TO CS1
024424  012777  000000  155644          MOV     #0,@A.CS1               ;CANCEL WRITE COMMAND
024432  017737  155652  002736          MOV     @A.IS,FP.ACT            ;READ IS REGISTER CONTENTS
024440  012737  000200  002734          MOV     #200,FP.EXP             ;LOAD EXPECTED VALUE:  IRY

024446  032737  100000  002736          BIT     #IS.ATA,FP.ACT          ;TEST FOR ATA RESET
024454  001405                          BEQ     200$                    ;CONTINUE IF ZERO


                                                                        ;''ATA NOT RESET BY LOADING -
024456                                  ERRHRD  7,M.ATA6,ER.DMP         ; VALID DATA TRANSFER COMMAND''
024456  104456                          TRAP    C$ERHRD
024460  000007                          .WORD   7
024462  007135                          .WORD   M.ATA6
024464  015074                          .WORD   ER.DMP
024466  000410                          BR      400$                    ;EXIT SUBTEST

024470  023737  002736  002734  200$:   CMP     FP.ACT,FP.EXP           ;COMPARE STATUS REGISTER VALUES
024476  001404                          BEQ     400$                    ;BRANCH IF IDENTICAL

024500                                  ERRHRD  8,M.ISR1,ER.DMP         ;''INTERFACE STATUS ERROR''
024500  104456                          TRAP    C$ERHRD
024502  000010                          .WORD   8
024504  011543                          .WORD   M.ISR1
024506  015074                          .WORD   ER.DMP

024510                          400$:   ENDSUB
024510                          L10060:
024510  104403                          TRAP    C$ESUB
```

```
 4601 024512  004737  015550                JSR     PC,CLRTST               ;AND CLEAR TEST DATA
 4602
 4603 024516                                ENDTST
      024516                        L10054:
      024516  104401                TRAP    C$ETST
```

```
                                .SBTTL  TEST 10:  CHECK WORD TRANSFER LOGIC
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK IRY AND CYC BITS FOR PROPER FUNCTION
                                ;;      AROUND DATA TRANSFER COMMANDS
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT BYTE MODE THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR STATUS
                                ;;          EXECUTE SUBTEST 1-3 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      INTERFACE STATUS IRY BIT NOT SET/RESET
                                ;;      MASSBUS CONTROLLER RDY BIT NOT SET/RESET
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T10::
                                SELTEST =       B.BYTE

024520  013700  003010        50$:    MOV     SELECT,R0               ;LOAD TEST SELECT WORD
024524  032700  000010                BIT     #SELTEST,R0             ;TEST BYTE MODE
024530  001402                        BEQ     75$                     ;CONTINUE IF DEFAULT

024532                                EXIT    TST                     ;NO:  EXIT TEST
024532  104432                        TRAP    C$EXIT
024534  000302                        .WORD   L10061-.

024536  004737  015342        75$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
024542  005737  003030                TST     MBCFLG                  ;TEST FLAG FOR ERROR
024546  001405                        BEQ     100$                    ;CONTINUE IF NO ERROR

024550                                ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
024550  104455                        TRAP    C$ERDF
024552  000001                        .WORD   1
024554  011650                        .WORD   M.MBC1
024556  015074                        .WORD   ER.DMP
024560                                DOCLN                           ;CLEAN UP AND EXIT
024560  104444                        TRAP    C$DCLN

024562  004737  015274        100$:   JSR     PC,SETISR               ;SETUP SERVICE ROUTINE VECTOR
                                                                      ;EXECUTE SUBTEST
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      SET CYCLE REQUEST DELAY A FEW CYCLES
                                        ;;      READ REGISTERS, CHECK IRY BIT SET AND RDY BIT SET
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB                          ;CHECK INTERFACE CLEAR CMD
                                        T10.1:
                                                TRAP    C$BSUB
                                        100$:   MOV     #IS.CYC,@A.IS           ;ISSUE CYCLE REQUEST
                                                JSR     PC,DLY                  ;DELAY A FEW CYCLES
                                                JSR     PC,RDREG                ;READ REGISTERS
                                                BIT     #IS.IRY,R.IS            ;TEST FOR IRY SET
                                                BNE     200$                    ;IRY SET:  CONTINUE
                                                ERRHRD  2,M.IRY2,ER.DMP         ;''IRY BIT RESET WHEN CYC BIT SET''
                                                TRAP    C$ERHRD
                                                .WORD   2
                                                .WORD   M.IRY2
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT FROM SUB TEST

                                        200$:   BIT     #CS1.RDY,R.CS1          ;TEST FOR RDY BIT SET
                                                BNE     400$                    ;RDY SET:  CONTINUE

                                                ERRHRD  3,M.RDY2,ER.DMP         ;'MBC RDY BIT NOT SET BY IRY''
                                                TRAP    C$ERHRD
                                                .WORD   3
                                                .WORD   M.RDY2
                                                .WORD   ER.DMP

                                        400$:   ENDSUB
                                        L10062:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      INITIALIZE REGISTERS AND ISSUE INTERFACE WRITE
                                        ;;      READ REGISTERS, CHECK IRY RESET AND RDY RESET
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB
                                        T10.2:
                                                TRAP    C$BSUB
                                        100$:   MOV     #-1,R0                  ;INITIALIZE BLOCK SIZE
                                                JSR     PC,WRTINI               ;AND SETUP ADDRESS REGISTERS
                                                MOV     #WR.CMD,@A.CS1          ;ISSUE READ TO INTERFACE

                                                JSR     PC,RDREG                ;READ REGISTERS INTO R.TBL
                                                BIT     #IS.IRY,R.IS            ;AND TEST FOR IRY RESET
                                                BEQ     200$                    ;IRY RESET:  CONTINUE

                                                ERRHRD  4,M.IRY3,ER.DMP         ;''IRY NOT RESET BY DATA TRANSFER CMD''
                                                TRAP    C$ERHRD
                                                .WORD   4
                                                .WORD   M.IRY3
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT FROM SUB TEST

                                        200$:   BIT     #CS1.RDY,R.CS1          ;TEST FOR RDY RESET
                                                BEQ     400$                    ;RDY RESET:  CONTINUE

                                                ERRHRD  5,M.RDY1,ER.DMP         ;'MBC RDY NOT RESET BY IRY''
                                                TRAP    C$ERHRD
                                                .WORD   5
                                                .WORD   M.RDY1
                                                .WORD   ER.DMP

                                        400$:   ENDSUB
                                        L10063:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 3:
                                        ;;
                                        ;;      SET CYCLE REQUEST AND DELAY FOR TWICE THE TRANSFER INTERVAL
                                        ;;      READ REGISTERS, CHECK IRY BIT SET AND RDY BIT SET
                                        ;;
                                        STARS
                                        ;;*****************************************************************
                                            SGNSUB
                                        T10.3:
                                            TRAP    C$BSUB
                                        100$:   MOV     #IS.CYC,@A.IS           ;ISSUE CYCLE REQUEST AND WAIT
                                            JSR     PC,TSTINT               ;TEST FOR INTERRUPT OR TIME OUT

                                            JSR     PC,RDREG                ;READ REGISTERS
                                            BIT     #IS.IRY,R.IS            ;TEST FOR IRY SET
                                            BNE     200$                    ;IRY SET:  CONTINUE

                                            ERRHRD  6,M.IRY4,ER.DMP         ;''IRY NOT SET BY WORD XFER EBL''
                                            TRAP    C$ERHRD
                                            .WORD   6
                                            .WORD   M.IRY4
                                            .WORD   ER.DMP
                                            BR      400$                    ;EXIT SUB TEST

                                        200$:   BIT     #IS.IRY,R.CS1           ;TEST FOR RDY SET
                                            BNE     400$                    ;RDY SET:  CONTINUE

                                            ERRHRD  7,M.RDY2,ER.DMP         ;'MBC RDY BIT NOT SET BY IRY''
                                            TRAP    C$ERHRD
                                            .WORD   7
                                            .WORD   M.RDY2
                                            .WORD   ER.DMP

                                        400$:   ENDSUB
                                        L10064:
                                            TRAP    C$ESUB

                                            JSR     PC,CLRISR               ;CLEAR VECTOR

                                            JSR     PC,CLRTST               ;AND CLEAR TEST DATA

                                            ENDTST
                                        L10061:
                                            TRAP    C$ETST
```

```
                                .SBTTL  TEST 11:  CHECK INTERRUPT ON WORD TRANSFER
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK THAT INTERRUPT WAS RECEIVED ON DATA TRANSFER COMPLETION
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT BYTE MODE THEN:
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR AND INIT BLOCK SIZE
                                ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                ;;
                                ;;  ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER CLEAR ERROR
                                ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T11::

                        000010  SELTEST =       B.BYTE

025040  013700  003010          50$:    MOV     SELECT,R0               ;LOAD TEST SELECT WORD
025044  032700  000010                  BIT     #SELTEST,R0             ;TEST BYTE MODE
025050  001402                          BEQ     75$                     ;CONTINUE IF NOT BYTE MODE

025052                                  EXIT    TST                     ;NO:  EXIT TEST
025052  104432                          TRAP    C$EXIT
025054  000242                          .WORD   L10065-.

025056  004737  015342          75$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
025062  005737  003030                  TST     MBCFLG                  ;TEST FLAG FOR ERROR
025066  001405                          BEQ     100$                    ;CONTINUE IF NO ERROR

025070                                  ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
025070  104455                          TRAP    C$ERDF
025072  000001                          .WORD   1
025074  011650                          .WORD   M.MBC1
025076  015074                          .WORD   ER.DMP
025100                                  DOCLN                           ;CLEAN UP AND EXIT
025100  104444                          TRAP    C$DCLN

025102  004737  015274          100$:   JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      WRITE BLOCK TO INTERFACE
                                        ;;      CHECK FOR INTERRUPT RECEIVED
                                        ;;
                                        STARS
                                        ;;*****************:***********************************************
                                                BGNSUB
                                        T11.1:
                                                TRAP    C$BSUB
                                                MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 WORDS
                                                JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
                                                JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
                                                TST     MBCFLG                  ;TEST FOR ERROR
                                                BEQ     150$                    ;BRANCH IF NO ERROR

                                                ERRHRD  2,M.WRD2,ER.DMP         ;''XFER ERROR ON INTERFACE WRITE''
                                                TRAP    C$ERHRD
                                                .WORD   2
                                                .WORD   M.WRD2
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT SUBTEST

                                        150$:   BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SE
                                                BNE     200$                    ;IRY SET:  CONTINUE

                                                ERRHRD  3,M.WRD1,ER.DMP         ;''DATA TRANSFER NOT COMPLETE''
                                                TRAP    C$ERHRD
                                                .WORD   3
                                                .WORD   M.WRD1
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT SUB TEST

                                        200$:   TST     INTFLG                  ;TEST FOR INTERRUPT RECEIVED
                                                BNE     400$                    ;BRANCH IF NOT ZERO

                                                ERRHRD  4,M.INT1,ER.DMP         ;''NO INTERRUPT ON WORD XFER EBL''
                                                TRAP    C$ERHR'
                                                .WORD   4
                                                .WORD   M.INT1
                                                .WORD   ER.DMP

                                        400$:   ENDSUB                          ;WRITE BLOCK
                                        L10066:
                                                TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SUBTEST 2:
                                        ;;
                                        ;;        READ BLOCK FROM INTERFACE
                                        ;;        CHECK FOR INTERRUPT RECEIVED
                                        ;;
                                        STARS
                                        ;;*****************************************************************
                                                BGNSUB
                                        T11.2:
                                                TRAP    C$BSUB
                                                MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 WORDS
                                                JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
                                                JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
                                                TST     MBCFLG                  ;TEST FOR ERROR
                                                BEQ     150$                    ;BRANCH IF NO ERROR

                                                ERRHRD  5,M.WRD3,ER.DMP         ;''XFER ERROR ON INTERFACE READ''
                                                TRAP    C$ERHRD
                                                .WORD   5
                                                .WORD   M.WRD3
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT SUBTEST

                                        150$:   BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
                                                BNE     200$                    ;IRY SET:  CONTINUE

                                                ERRHRD  6,M.WRD1,ER.DMP         ;''DATA TRANSFER NOT COMPLETE''
                                                TRAP    C$ERHRD
                                                .WORD   6
                                                .WORD   M.WRD1
                                                .WORD   ER.DMP
                                                BR      400$                    ;EXIT SUB TEST

                                        200$:   TST     INTFLG                  ;TEST FOR INTERRUPT RECEIVED
                                                BNE     400$                    ;BRANCH IF NOT ZERO

                                                ERRHRD  7,M.INT1,ER.DMP         ;''NO INTERRUPT ON WORD XFER EBL''
                                                TRAP    C$ERHRD
                                                .WORD   7
                                                .WORD   M.INT1
                                                .WORD   ER.DMP

                                        400$:   ENDSUB                          ;READ BLOCK
                                        L10067:
                                                TRAP    C$ESUB

                                                JSR     PC,CLRISR               ;AND RESTORE VECTOR ADDRESS

                                                JSR     PC,CLRTST               ;AND CLEAR TEST DATA

                                                ENDTST
                                        L10065:
                                                TRAP    C$ETST
```

```
                                .SBTTL  TEST 12:  CHECK WORD BLOCK DATA TRANSFER
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      PERFORM 64-WORD DMA DATA TRANSFERS,
                                ;;      COMPARE DATA READ AGAINST DATA WRITTEN
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT BYTE MODE THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          INITIALIZE TRANSFER PARAMETERS AND
                                ;;          EXECUTE SUBTEST 1-4 AND CLEAR INTERRUPT VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      DATA TRANSFER NOT COMPLETE
                                ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                ;;      DATA RECEIVED DOES NOT MATCH EXPECTED DATA
                                ;;
                                ;;    PROMPTS:
                                ;;      CHECK THAT SW1-7 IS IN THE 'ON' POSITION.
                                STARS
                                ;;*****************************************************************

                                        BGNTST
025320                          T12::
                                SELTEST =       B.BYTE
025320  013700  003010                  MOV     SELECT,R0               ;READ SELECT WORD
025324  032700  000010                  BIT     #SELTEST,R0             ;AND TEST BYTE MODE BIT SET
025330  001402                          BEQ     50$                     ;BRANCH IF NOT SET
025332                                  EXIT    TST                     ;BYTE MODE:  EXIT TEST
025332  104432                          TRAP    C$EXIT
025334  002176                          .WORD   L10070-.
025336  004737  015274          50$:    JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
025342  012737  177700  002506          MOV     #-100,D.BLOCK           ;INITIALIZE BLOCK SIZE
025350  012737  000200  002512          MOV     #200,D.MAX              ;INITIALIZE MAX DATA BUFFER INDEX
025356  012737  000001  002510          MOV     #1,D.LPCNT              ;INITIALIZE LOOP COUNTER
025364                          LP.WXFR:                                ;TRANSFER LOOP LABEL
```

```
 4883 025364                            STARS
                                        ;;*****************************************************************
 4884                                   ;;
 4885                                   ;;    SUBTEST 1:
 4886                                   ;;
 4887                                   ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 4888                                   ;;      SETUP DATA BUFFER FOR ALL ONE'S
 4889                                   ;;
 4890                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 4891                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 4892                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 4893                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 4894                                   ;;
 4895                                   ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 4896                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 4897                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 4898                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 4899                                   ;;
 4900                                   ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 4901 025364                            STARS
                                        ;;*****************************************************************
 4902
 4903 025364                                    BGNSUB
      025364                            T12.1:
      025364  104402                            TRAP    C$BSUB
 4904
 4905 025366  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 4906 025372  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 4907 025376  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 4908
 4909 025400                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      025400  104455                            TRAP    C$ERDF
      025402  000001                            .WORD   1
      025404  011650                            .WORD   M.MBC1
      025406  015074                            .WORD   ER.DMP
 4910 025410                                    DOCLN                           ;CLEAN UP AND EXIT
      025410  104444                            TRAP    C$DCLN
 4911
 4912 025412  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 4913
 4914 025414  012762  177777  004054    150$:   MOV     #-1,OBUF(R2)            ;LOAD ONES INTO BUFFER
 4915 025422  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 4916 025430  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 4917 025434  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 4918 025440  003365                            BGT     150$                    ;CONTINUE UNTIL BUFFER INITIALIZED
 4919
 4920 025442  012700  177700            200$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
 4921 025446  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 4922 025452  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 4923 025456  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
 4924 025462  001005                            BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 4925
 4926 025464                                    ERRHRD  2,M.INT1,ER.DMP         ;"NOT INTERRUPT ON WORD XFER EBL"
      025464  104456                            TRAP    C$ERHRD
      025466  000002                            .WORD   2
      025470  010706                            .WORD   M.INT1
      025472  015074                            .WORD   ER.DMP
```

```
4927 025474  000407                          BR      300$                    ;EXIT SEGMENT
4928
4929 025476  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
4930 025502  001404                          BEQ     300$                    ;CONTINUE IF RESET
4931
4932 025504                                  ERRHRD  3,M.WRD2,ER.DMP         ;'WORD XFER ERROR ON INTERFACE WRITE''
     025504  104456                          TRAP    C$ERHRD
     025506  000003                          .WORD   3
     025510  007324                          .WORD   M.WRD2
     025512  015074                          .WORD   ER.DMP
4933
4934 025514  012700  177700          300$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
4935 025520  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
4936 025524  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
4937 025530  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
4938 025534  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
4939
4940 025536                                  ERRHRD  4,M.INT1,ER.DMP         ;'NOT INTERRUPT ON WORD XFER EBL''
     025536  104456                          TRAP    C$ERHRD
     025540  000004                          .WORD   4
     025542  010706                          .WORD   M.INT1
     025544  015074                          .WORD   ER.DMP
4941 025546  000407                          BR      400$                    ;EXIT SEGMENT
4942
4943 025550  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
4944 025554  001404                          BEQ     400$                    ;CONTINUE IF RESET
4945
4946 025556                                  ERRHRD  5,M.WRD3,ER.DMP         ;'WORD XFER ERROR ON INTERFACE READ''
     025556  104456                          TRAP    C$ERHRD
     025560  000005                          .WORD   5
     025562  007373                          .WORD   M.WRD3
     025564  015074                          .WORD   ER.DMP
```

```
 4948
 4949 025566  005002                    400$:  CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 4950 025570  005037  003026                   CLR     ERRFLG                  ;CLEAR ERROR COUNT
 4951
 4952 025574  026262  003054  004054    425$:  CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
 4953 025602  001431                           BEQ     475$                    ;IDENTICAL:  CONTINUE
 4954
 4955 025604  005737  003026                   TST     ERRFLG                  ;TEST ERROR COUNT
 4956 025610  001006                           BNE     450$                    ;BRANCH IF NOT FIRST ERROR
 4957
 4958 025612                                   ERRHRD  6,M.WRD4,ER.DATA        ;'DATA COMPARISON ERROR''
      025612  104456                           TRAP    C$ERHRD
      025614  000006                           .WORD   6
      025616  007441                           .WORD   M.WRD4
      025620  015172                           .WORD   ER.DATA
 4959 025622  010237  003002                   MOV     R2,ERRNDX               ;SAVE DATA ERROR INDEX
 4960
 4961 025626  005237  003026            450$:  INC     ERRFLG                  ;BUMP ERROR COUNT
 4962 025632  022737  000010  003026           CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
 4963 025640  002412                           BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
 4964
 4965 025642  010237  002742                   MOV     R2,FP.NDX               ;SAVE DATA INDEX
 4966 025646  016237  003054  002736           MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
 4967 025654  016237  004054  002734           MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
 4968 025662  004737  016374                   JSR     PC,DSPDATA              ;AND DISPLAY DATA
 4969
 4970 025666  062702  000002            475$:  ADD     #2,R2                   ;BUMP INDEX
 4971 025672  023702  002512                   CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 4972 025676  003336                           BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
 4973
 4974 025700  005737  003026            500$:  TST     ERRFLG                  ;TEST NUMBER ERRORS
 4975 025704  001434                           BEQ     600$                    ;BRANCH IF NO ERRORS
 4976
 4977 025706                                   PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
      025706  013746  003026                   MOV     ERRFLG,-(SP)
      025712  012746  013304                   MOV     #F.ERCNT,-(SP)
      025716  012746  000002                   MOV     #2,-(SP)
      025722  010600                           MOV     SP,R0
      025724  104415                           TRAP    C$PNTX
      025726  062706  000006                   ADD     #6,SP
 4978 025732  005237  003004                   INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
 4979
 4980 025736  022737  000040  003026           CMP     #40,ERRFLG              ;TEST FOR 32 COMPARISON ERRORS
 4981 025744  001014                           BNE     600$                    ;NO:  EXIT SUBTEST
 4982 025746  022737  000100  003002           CMP     #100,ERRNDX             ;TEST FOR 33RD WORD IN ERROR
 4983 025754  001010                           BNE     600$                    ;NO:  EXIT SUBTEST
 4984
 4985 025756                                   PRINTB  #F.MODE                 ;PROMPT OPERATOR TO CHECK SW1-7
      025756  012746  013340                   MOV     #F.MODE,-(SP)
      025762  012746  000001                   MOV     #1,-(SP)
      025766  010600                           MOV     SP,R0
      025770  104414                           TRAP    C$PNTB
      025772  062706  000004                   ADD     #4,SP
 4986
 4987 025776                            600$:  ENDSUB
      025776                            L10071:
      025776  104403                           TRAP    C$ESUB
```

```
 4989 026000                            STARS
                                        ;;*****************************************************************
 4990                                   ;;
 4991                                   ;;    SUBTEST 2:
 4992                                   ;;
 4993                                   ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 4994                                   ;;      SETUP DATA BUFFER FOR ALL ZERO'S
 4995                                   ;;
 4996                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 4997                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 4998                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 4999                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5000                                   ;;
 5001                                   ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 5002                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 5003                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 5004                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5005                                   ;;
 5006                                   ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 5007 026000                            STARS
                                        ;;*****************************************************************
 5008 026000                                    BGNSUB
      026000                            T12.2:
      026000  104402                            TRAP    C$BSUB
 5009
 5010 026002  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 5011 026006  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 5012 026012  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 5013
 5014 026014                                    ERRDF   7,M.MBC1,ER.DMP         ;DEVICE FATAL
      026014  104455                            TRAP    C$ERDF
      026016  000007                            .WORD   7
      026020  011650                            .WORD   M.MBC1
      026022  015074                            .WORD   ER.DMP
 5015 026024                                    DOCLN                           ;CLEAN UP AND EXIT
      026024  104444                            TRAP    C$DCLN
 5016
 5017 026026  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 5018
 5019 026030  012762  000000  004054    150$:   MOV     #0,OBUF(R2)             ;LOAD ZERO'S INTO BUFFER
 5020 026036  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 5021 026044  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 5022 026050  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 5023 026054  003365                            BGT     150$                    ;CONTINUE UNTIL BUFFER INITIALIZED
 5024
 5025 026056  012700  177700            200$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
 5026 026062  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 5027 026066  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 5028 026072  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
 5029 026076  001005                            BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 5030
 5031 026100                                    ERRHRD  8,M.INT1,ER.DMP         ;"NO INTERRUPT ON WORD XFER EBL"
      026100  104456                            TRAP    C$ERHRD
      026102  000010                            .WORD   8
      026104  010706                            .WORD   M.INT1
      026106  015074                            .WORD   ER.DMP
 5032 026110  000407                            BR      300$                    ;EXIT SEGMENT
```

```
5033
5034 026112  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
5035 026116  001404                          BEQ     300$                    ;CONTINUE IF RESET
5036
5037 026120                                  ERRHRD  9,M.WRD2,ER.DMP         ;'WORD XFER ERROR ON INTERFACE WRITE''
     026120  104456                          TRAP    C$ERHRD
     026122  000011                          .WORD   9
     026124  007324                          .WORD   M.WRD2
     026126  015074                          .WORD   ER.DMP
5038
5039 026130  012700  177700          300$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
5040 026134  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
5041 026140  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
5042 026144  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
5043 026150  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
5044
5045 026152                                  ERRHRD  10,M.INT1,ER.DMP        ;'NO INTERRUPT ON WORD XFER EBL''
     026152  104456                          TRAP    C$ERHRD
     026154  000012                          .WORD   10
     026156  010706                          .WORD   M.INT1
     026160  015074                          .WORD   ER.DMP
5046 026162  000407                          BR      400$                    ;EXIT SEGMENT
5047
5048 026164  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
5049 026170  001404                          BEQ     400$                    ;CONTINUE IF RESET
5050
5051 026172                                  ERRHRD  11,M.WRD3,ER.DMP        ;'WORD XFER ERROR ON INTERFACE READ''
     026172  104456                          TRAP    C$ERHRD
     026174  000013                          .WORD   11
     026176  007373                          .WORD   M.WRD3
     026200  015074                          .WORD   ER.DMP
```

```
5053
5054 026202  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
5055 026204  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
5056
5057 026210  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
5058 026216  001431                            BEQ     475$                    ;IDENTICAL:  CONTINUE
5059
5060 026220  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
5061 026224  001006                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
5062
5063 026226                                    ERRHRD  12,M.WRD4,ER.DATA       ;''DATA COMPARISON ERROR''
     026226  104456                            TRAP    C$ERHRD
     026230  000014                            .WORD   12
     026232  007441                            .WORD   M.WRD4
     026234  015172                            .WORD   ER.DATA
5064 026236  010237  003002                    MOV     R2,ERRNDX               ;SAVE DATA ERROR INDEX
5065
5066 026242  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
5067 026246  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
5068 026254  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
5069
5070 026256  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
5071 026262  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
5072 026270  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
5073 026276  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
5074
5075 026302  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
5076 026306  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
5077 026312  003336                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
5078
5079
5080 026314  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
5081 026320  001434                            BEQ     600$                    ;BRANCH IF NO ERRORS
5082
5083 026322                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     026322  013746  003026                    MOV     ERRFLG,-(SP)
     026326  012746  013304                    MOV     #F.ERCNT,-(SP)
     026332  012746  000002                    MOV     #2,-(SP)
     026336  010600                            MOV     SP,R0
     026340  104415                            TRAP    C$PNTX
     026342  062706  000006                    ADD     #6,SP
5084 026346  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
5085
5086 026352  022737  000040  003026            CMP     #40,ERRFLG              ;TEST FOR 32 COMPARISON ERRORS
5087 026360  001014                            BNE     600$                    ;NO:  EXIT SUBTEST
5088 026362  022737  000100  003002            CMP     #100,ERRNDX             ;TEST FOR 33RD WORD IN ERROR
5089 026370  001010                            BNE     600$                    ;NO:  EXIT SUBTEST
5090
5091 026372                                    PRINTB  #F.MODE                 ;PROMPT OPERATOR TO CHECK SW1-7
     026372  012746  013340                    MOV     #F.MODE,-(SP)
     026376  012746  000001                    MOV     #1,-(SP)
     026402  010600                            MOV     SP,R0
     026404  104414                            TRAP    C$PNTB
     026406  062706  000004                    ADD     #4,SP
5092
5093 026412                            600$:   ENDSUB
     026412                            L10072:
```

TEST 12: CHECK WORD BLOCK DATA TRANSFER

```
        026412  104403                          TRAP    C$ESUB
```

```
 026414                                   STARS
                                          ;;*************************************************************
                                          ;;
                                          ;;          SUBTEST 3:
                                          ;;
                                          ;;            CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
                                          ;;            SETUP DATA BUFFER WITH MOVING ONE PATTERN
                                          ;;
                                          ;;            INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
                                          ;;            SET CYCLE REQUEST, DELAY AND READ REGISTERS
                                          ;;            CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
                                          ;;            IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
                                          ;;
                                          ;;            INITIALIZE REGISTERS FOR READ W/INT ENABLE
                                          ;;            SET CYCLE REQUEST, DELAY AND READ REGISTERS
                                          ;;            CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
                                          ;;            IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
                                          ;;
                                          ;;            COMPARE DATA READ AGAINST DATA WRITTEN
 026414                                   STARS
                                          ;;*************************************************************
 026414                                           BGNSUB                          ;MOVING ONE PATTERN
 026414                                   T12.3:
 026414  104402                                   TRAP    C$BSUB

 026416  004737  015342                           JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 026422  005737  003030                           TST     MBCFLG                  ;TEST FLAG FOR ERROR
 026426  001405                                   BEQ     100$                    ;CONTINUE IF NO ERROR

 026430                                           ERRDF   13,M.MBC1,ER.DMP        ;DEVICE FATAL
 026430  104455                                   TRAP    C$ERDF
 026432  000015                                   .WORD   13
 026434  011650                                   .WORD   M.MBC1
 026436  015074                                   .WORD   ER.DMP
 026440                                           DOCLN                           ;CLEAN UP AND EXIT
 026440  104444                                   TRAP    C$DCLN

 026442  005002                           100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 026444  012705  000004                           MOV     #4,R5                   ;INITIALIZE COUNTER

 026450  012704  000001                   125$:   MOV     #1,R4                   ;INITIALIZE PATTERN REGISTER

 026454  010462  004054                   150$:   MOV     R4,OBUF(R2)             ;LOAD ONES INTO BUFFER
 026460  012762  000000  003054                   MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 026466  062702  000002                           ADD     #2,R2                   ;AND BUMP INDEX
 026472  023702  002512                           CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 026476  001404                                   BEQ     200$                    ;BRANCH IF END OF BUFFER

 026500  006304                                   ASL     R4                      ;SHIFT PATTERN
 026502  001364                                   BNE     150$                    ;NO:  CONTINUE
 026504  005305                                   DEC     R5                      ;DECREMENT BLOCK COUNTER
 026506  001360                                   BNE     125$                    ;AND CONTINUE IF ANOTHER BLOCK

 026510  012700  177700                   200$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
 026514  004737  015674                           JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 026520  004737  015410                           JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 026524  005737  003022                           TST     INTFLG                  ;TEST FOR INTERRUPT
```

```
5143 026530  001005                          BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
5144
5145 026532                                  ERRHRD  14,M.INT1,ER.DMP        ;'NO INTERRUPT ON WORD XFER EBL''
     026532  104456                          TRAP    C$ERHRD
     026534  000016                          .WORD   14
     026536  010706                          .WORD   M.INT1
     026540  015074                          .WORD   ER.DMP
5146 026542  000407                          BR      300$                    ;EXIT SEGMENT
5147
5148 026544  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
5149 026550  001404                          BEQ     300$                    ;CONTINUE IF RESET
5150
5151 026552                                  ERRHRD  15,M.WRD2,ER.DMP        ;'WORD XFER ERROR ON INTERFACE WRITE''
     026552  104456                          TRAP    C$ERHRD
     026554  000017                          .WORD   15
     026556  007324                          .WORD   M.WRD2
     026560  015074                          .WORD   ER.DMP
5152
5153 026562  012700  177700          300$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
5154 026566  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
5155 026572  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
5156 026576  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
5157 026602  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
5158
5159 026604                                  ERRHRD  16,M.INT1,ER.DMP        ;'NO INTERRUPT ON WORD XFER EBL''
     026604  104456                          TRAP    C$ERHRD
     026606  000020                          .WORD   16
     026610  010706                          .WORD   M.INT1
     026612  015074                          .WORD   ER.DMP
5160 026614  000407                          BR      400$                    ;EXIT SEGMENT
5161
5162 026616  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
5163 026622  001404                          BEQ     400$                    ;CONTINUE IF RESET
5164
5165 026624                                  ERRHRD  17,M.WRD3,ER.DMP        ;'WORD XFER ERROR ON INTERFACE READ''
     026624  104456                          TRAP    C$ERHRD
     026626  000021                          .WORD   17
     026630  007373                          .WORD   M.WRD3
     026632  015074                          .WORD   ER.DMP
```

```
 5167
 5168 026634  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 5169 026636  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
 5170
 5171 026642  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
 5172 026650  001431                            BEQ     475$                    ;IDENTICAL:  CONTINUE
 5173
 5174 026652  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
 5175 026656  001006                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
 5176
 5177 026660                                    ERRHRD  18,M.WRD4,ER.DATA       ;''DATA COMPARISON ERROR''
      026660  104456                            TRAP    C$ERHRD
      026662  000022                            .WORD   18
      026664  007441                            .WORD   M.WRD4
      026666  015172                            .WORD   ER.DATA
 5178 026670  010237  003002                    MOV     R2,ERRNDX               ;SAVE DATA ERROR INDEX
 5179
 5180 026674  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
 5181 026700  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
 5182 026706  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
 5183
 5184 026710  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
 5185 026714  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
 5186 026722  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
 5187 026730  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
 5188
 5189 026734  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
 5190 026740  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 5191 026744  003336                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
 5192
 5193
 5194 026746  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
 5195 026752  001434                            BEQ     600$                    ;BRANCH IF NO ERRORS
 5196
 5197 026754                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
      026754  013746  003026                    MOV     ERRFLG,-(SP)
      026760  012746  013304                    MOV     #F.ERCNT,-(SP)
      026764  012746  000002                    MOV     #2,-(SP)
      026770  010600                            MOV     SP,R0
      026772  104415                            TRAP    C$PNTX
      026774  062706  000006                    ADD     #6,SP
 5198 027000  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
 5199
 5200 027004  022737  000040  003026            CMP     #40,ERRFLG              ;TEST FOR 32 COMPARISON ERRORS
 5201 027012  001014                            BNE     600$                    ;NO:  EXIT SUBTEST
 5202 027014  022737  000100  003002            CMP     #100,ERRNDX             ;TEST FOR 33RD WORD IN ERROR
 5203 027022  001010                            BNE     600$                    ;NO:  EXIT SUBTEST
 5204
 5205 027024                                    PRINTB  #F.MODE                 ;PROMPT OPERATOR TO CHECK SW1-7
      027024  012746  013340                    MOV     #F.MODE,-(SP)
      027030  012746  000001                    MOV     #1,-(SP)
      027034  010600                            MOV     SP,R0
      027036  104414                            TRAP    C$PNTB
      027040  062706  000004                    ADD     #4,SP
 5206
 5207 027044                            600$:   ENDSUB
      027044                            L10073:
```

```
ZDRMAO DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 149-1
TEST 12:  CHECK WORD BLOCK DATA TRANSFER

        027044  104403                          TRAP    C$ESUB
```

```
 5209 027046                                    STARS
                                                ;;*****************************************************************
 5210                                           ;;
 5211                                           ;;    SUBTEST 4:
 5212                                           ;;
 5213                                           ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 5214                                           ;;      SETUP DATA BUFFER WITH MOVING ZERO PATTERN
 5215                                           ;;
 5216                                           ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 5217                                           ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 5218                                           ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 5219                                           ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5220                                           ;;
 5221                                           ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 5222                                           ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 5223                                           ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 5224                                           ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5225                                           ;;
 5226                                           ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 5227 027046                                    STARS
                                                ;;*****************************************************************
 5228 027046                                            BGNSUB                          ;MOVING ZERO PATTERN
      027046                                    T12.4:
      027046  104402                                    TRAP    C$BSUB
 5229
 5230 027050  004737  015342                            JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 5231 027054  005737  003030                            TST     MBCFLG                  ;TEST FLAG FOR ERROR
 5232 027060  001405                                    BEQ     100$                    ;CONTINUE IF NO ERROR
 5233
 5234 027062                                            ERRDF   19,M.MBC1,ER.DMP        ;DEVICE FATAL
      027062  104455                                    TRAP    C$ERDF
      027064  000023                                    .WORD   19
      027066  011650                                    .WORD   M.MBC1
      027070  015074                                    .WORD   ER.DMP
 5235 027072                                            DOCLN                           ;CLEAN UP AND EXIT
      027072  104444                                    TRAP    C$DCLN
 5236
 5237 027074  005002                        100$:       CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 5238 027076  012705  000004                            MOV     #4,R5                   ;INITIALIZE COUNTER
 5239
 5240 027102  012704  177776                125$:       MOV     #-2,R4                  ;INITIALIZE PATTERN REGISTER
 5241
 5242 027106  010462  004054                150$:       MOV     R4,OBUF(R2)             ;LOAD ONES INTO BUFFER
 5243 027112  012762  000000  003054                    MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 5244 027120  062702  000002                            ADD     #2,R2                   ;AND BUMP INDEX
 5245 027124  023702  002512                            CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 5246 027130  001410                                    BEQ     200$                    ;BRANCH IF END OF BUFFER
 5247
 5248 027132  006304                                    ASL     R4                      ;SHIFT PATTERN
 5249 027134  052704  000001                            BIS     #BIT00,R4               ;SET INCOMING BIT
 5250 027140  022704  177777                            CMP     #-1,R4                  ;TEST FOR ALL BITS ONE
 5251 027144  001360                                    BNE     150$                    ;NO:  CONTINUE
 5252 027146  005305                                    DEC     R5                      ;DECREMENT BLOCK COUNTER
 5253 027150  001354                                    BNE     125$                    ;AND CONTINUE IF ANOTHER BLOCK
 5254
 5255 027152  012700  177700                200$:       MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
 5256 027156  004737  015674                            JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
```

```
 5257 027162  004737  015410                JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 5258 027166  005737  003022                TST     INTFLG                  ;TEST FOR INTERRUPT
 5259 027172  001005                        BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 5260
 5261 027174                                ERRHRD  20,M.INT1,ER.DMP        ;'NO INTERRUPT ON WORD XFER EBL''
      027174  104456                        TRAP    C$ERHRD
      027176  000024                        .WORD   20
      027200  010706                        .WORD   M.INT1
      027202  015074                        .WORD   ER.DMP
 5262 027204  000407                        BR      300$                    ;EXIT SEGMENT
 5263
 5264 027206  005737  003030          250$: TST     MBCFLG                  ;TEST FOR ERROR
 5265 027212  001404                        BEQ     300$                    ;CONTINUE IF RESET
 5266
 5267 027214                                ERRHRD  21,M.WRD2,ER.DMP        ;'WORD XFER ERROR ON INTERFACE WRITE''
      027214  104456                        TRAP    C$ERHRD
      027216  000025                        .WORD   21
      027220  007324                        .WORD   M.WRD2
      027222  015074                        .WORD   ER.DMP
 5268
 5269 027224  012700  177700          300$: MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
 5270 027230  004737  015642                JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
 5271 027234  004737  015410                JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 5272 027240  005737  003022                TST     INTFLG                  ;TEST FOR INTERRUPT
 5273 027244  001005                        BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
 5274
 5275 027246                                ERRHRD  22,M.INT1,ER.DMP        ;'NO INTERRUPT ON WORD XFER EBL''
      027246  104456                        TRAP    C$ERHRD
      027250  000026                        .WORD   22
      027252  010706                        .WORD   M.INT1
      027254  015074                        .WORD   ER.DMP
 5276 027256  000407                        BR      400$                    ;EXIT SEGMENT
 5277
 5278 027260  005737  003030          350$: TST     MBCFLG                  ;TEST FOR ERROR
 5279 027264  001404                        BEQ     400$                    ;CONTINUE IF RESET
 5280
 5281 027266                                ERRHRD  23,M.WRD3,ER.DMP        ;'WORD XFER ERROR ON INTERFACE READ''
      027266  104456                        TRAP    C$ERHRD
      027270  000027                        .WORD   23
      027272  007373                        .WORD   M.WRD3
      027274  015074                        .WORD   ER.DMP
```

```
5283
5284 027276  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
5285 027300  005037  003026                     CLR     ERRFLG                  ;CLEAR ERROR COUNT
5286
5287 027304  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
5288 027312  001431                     425$:   BEQ     475$                    ;IDENTICAL:  CONTINUE
5289
5290 027314  005737  003026                     TST     ERRFLG                  ;TEST ERROR COUNT
5291 027320  001006                             BNE     450$                    ;BRANCH IF NOT FIRST ERROR
5292
5293 027322                                     ERRHRD  24,M.WRD4,ER.DATA       ;'DATA COMPARISON ERROR''
     027322  104456                             TRAP    C$ERHRD
     027324  000030                             .WORD   24
     027326  007441                             .WORD   M.WRD4
     027330  015172                             .WORD   ER.DATA
5294 027332  010237  003002                     MOV     R2,ERRNDX               ;SAVE DATA ERROR INDEX
5295
5296 027336  005237  003026             450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
5297 027342  022737  000010  003026             CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
5298 027350  002412                             BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
5299
5300 027352  010237  002742                     MOV     R2,FP.NDX               ;SAVE DATA INDEX
5301 027356  016237  003054  002736             MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
5302 027364  016237  004054  002734             MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
5303 027372  004737  016374                     JSR     PC,DSPDATA              ;AND DISPLAY DATA
5304
5305 027376  062702  000002             475$:   ADD     #2,R2                   ;BUMP INDEX
5306 027402  023702  002512                     CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
5307 027406  003336                             BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
5308
5309 027410  005737  003026             500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
5310 027414  001434                             BEQ     600$                    ;BRANCH IF NO ERRORS
5311
5312 027416                                     PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     027416  013746  003026                     MOV     ERRFLG,-(SP)
     027422  012746  013304                     MOV     #F.ERCNT,-(SP)
     027426  012746  000002                     MOV     #2,-(SP)
     027432  010600                             MOV     SP,R0
     027434  104415                             TRAP    C$PNTX
     027436  062706  000006                     ADD     #6,SP
5313 027442  005237  003004                     INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
5314
5315 027446  022737  000040  003026             CMP     #40,ERRFLG              ;TEST FOR 32 COMPARISON ERRORS
5316 027454  001014                             BNE     600$                    ;NO:  EXIT SUBTEST
5317 027456  022737  000100  003002             CMP     #100,ERRNDX             ;TEST FOR 33RD WORD IN ERROR
5318 027464  001010                             BNE     600$                    ;NO:  EXIT SUBTEST
5319
5320 027466                                     PRINTB  #F.MODE                 ;PROMPT OPERATOR TO CHECK SW1-7
     027466  012746  013340                     MOV     #F.MODE,-(SP)
     027472  012746  000001                     MOV     #1,-(SP)
     027476  010600                             MOV     SP,R0
     027500  104414                             TRAP    C$PNTB
     027502  062706  000004                     ADD     #4,SP
5321
5322 027506                             600$:   ENDSUB
     027506                             L10074:
     027506  104403                             TRAP    C$ESUB
```

```
ZDRMAO DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 151-1
TEST 12:  CHECK WORD BLOCK DATA TRANSFER

   5323
```

```
5325
5326 027510  005337  002510                  DEC     D.LPCNT                 ;DECREMENT LOOP COUNTER
5327 027514  001402                          BEQ     800$                    ;AND EXIT IF ZERO
5328 027516  000137  025364                  JMP     LP.WXFR                 ;CONTINUE TRANSFER
5329
5330 027522  004737  015324          800$:   JSR     PC,CLRISR               ;CLEAR INTERRUPT SERVICE VECTOR
5331
5332 027526  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
5333
5334 027532                                  ENDTST
     027532                          L10070:
     027532  104401                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 13:  CHECK WORD TRANSFER ABORT LOGIC (XABT)
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  TEST DESCRIPTION:
                                        ;;
                                        ;;      CHECK WORD MODE DATA TRANSFER ABORT LOGIC THROUGH XABT BIT
                                        ;;
                                        ;;    TEST STEPS:
                                        ;;
                                        ;;      IF NOT REV "A" AND NOT BYTE MODE THEN:
                                        ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                        ;;          EXECUTE SUBTEST 1-2 AND CLEAR INTERRUPT SERVICE VECTOR
                                        ;;
                                        ;;    ERROR CONDITIONS:
                                        ;;
                                        ;;      MASSBUS CONTROLLER CLEAR ERROR
                                        ;;      NO WRITE/READ TRANSFER ABORT
                                        ;;      TRE NOT SET:  ADJUST BANDWIDTH
                                        ;;      INTERFACE IRY BIT NOT SET BY TRANSFER ABORT
                                        STARS
                                        ;;*****************************************************************

027534                                          BGNTST
027534                                  T13::

000210                                  SELMASK =       B.REVA + B.BYTE

027534  013700  003010                          MOV     SELECT,R0               ;READ SELECT WORD
027540  032700  000210                          BIT     #SELMASK,R0             ;TEST FOR REVA OR BYTE BIT SET
027544  001402                                  BEQ     50$                     ;BRANCH IF RESET

027546                                          EXIT    TST                     ;REVA OR BYTE MODE:  EXIT TEST
027546  104432                                  TRAP    C$EXIT
027550  000462                                  .WORD   L10075-.

027552  004737  015274                  50$:    JSR     PC,SETISR               ;SETUP VECTOR
027556  013737  002406  002730                  MOV     NA.CS1,FP.NAM           ;LOAD REGISTER NAME ADDRESS
027564  013737  002436  002746                  MOV     BA.CS1,FP.TBL           ;LOAD REGISTER BIT TABLE ADDRESS
                                                                                ;EXECUTE SUBTEST
```

```
 5372 027572                            STARS
                                        ;;*****************************************************************
 5373                                   ;;
 5374                                   ;;    SUBTEST 1:
 5375                                   ;;
 5376                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 5377                                   ;;      ISSUE INTERFACE WRITE CMD AND SET CYCLE REQUEST BIT
 5378                                   ;;      DELAY A FEW CYCLES AND SET IS REGISTER XABT BIT
 5379                                   ;;      READ REGISTERS, CHECK IRY BIT SET
 5380                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
 5381 027572                            STARS
                                        ;;*****************************************************************
 5382
 5383 027572                                    BGNSUB
      027572                            T13.1:
      027572  104402                           TRAP    C$BSUB
 5384
 5385 027574  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 5386 027600  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 5387 027604  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 5388
 5389 027606                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      027606  104455                           TRAP    C$ERDF
      027610  000001                           .WORD   1
      027612  011650                           .WORD   M.MBC1
      027614  015074                           .WORD   ER.DMP
 5390 027616                                    DOCLN                           ;CLEAN UP AND EXIT
      027616  104444                           TRAP    C$DCLN
 5391
 5392 027620  012700  177600            100$:   MOV     #-200,R0                ;LOAD BLOCK SIZE = 128 WORDS
 5393 027624  004737  015762                    JSR     PC,WRTINI               ;AND INITIALIZE ADDRESS REGISTERS
 5394 027630  012777  000161  152440            MOV     #WIE.CMD,@A.CS1         ;ISSUE WRITE COMMAND TO CS1
 5395 027636  004737  016030                    JSR     PC,SILODLY              ;WAIT FOR SILO TO FILL
 5396 027642  012777  000020  152440            MOV     #IS.CYC,@A.IS           ;SET IS REGISTER CYC BIT
 5397 027650  004737  016016                    JSR     PC,DMADLY               ;DELAY A FEW CYCLES
 5398
 5399 027654  012777  004000  152426            MOV     #IS.XABT,@A.IS          ;SET IS REGISTER TRANSFER ABORT
 5400 027662  004737  015444                    JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME-OUT
 5401 027666  004737  015606                    JSR     PC,RDREG                ;READ REGISTERS
 5402
 5403 027672  032737  000200  002340            BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
 5404 027700  001005                            BNE     200$                    ;YES:  CONTINUE
 5405
 5406 027702                                    ERRHRD  2,M.IRY5,ER.DMP         ;"IRY NOT SET BY WORD TRANSFER ABORT"
      027702  104456                           TRAP    C$ERHRD
      027704  000002                           .WORD   2
      027706  011320                           .WORD   M.IRY5
      027710  015074                           .WORD   ER.DMP
 5407 027712  000425                            BR      400$                    ;EXIT SUBTEST
 5408
 5409 027714  005737  002330            200$:   TST     R.WC                    ;TEST WORD COUNT
 5410 027720  100404                            BMI     300$                    ;IF LESS THAN ZERO THEN OK
 5411
 5412 027722                                    ERRHRD  3,M.ABT1,ER.DMP         ;"NO XFER ABORT ON INTERFACE WRITE"
      027722  104456                           TRAP    C$ERHRD
      027724  000003                           .WORD   3
      027726  006375                           .WORD   M.ABT1
```

```
        027730  015074                          .WORD   ER.DMP
   5413
   5414 027732  012737  144260  002734  300$:   MOV     #144260,FP.EXP          ;EXPECTED VALUE:  SC,TRE,DVA,RDY,WRITE
   5415 027740  023737  002734  002326          CMP     FP.EXP,R.CS1            ;COMPARE WITH EXPECTED MBA STATUS
   5416 027746  001407                          BEQ     400$                    ;IF EQ THEN CONTINUE
   5417
   5418 027750  013737  002326  002736          MOV     R.CS1,FP.ACT            ;LOAD ACTUAL VALUE
   5419 027756                                  ERRHRD  4,M.CSR2,ER.REG         ;"CS1 REGISTER STATUS ERROR"
        027756  104456                          TRAP    C$ERHRD
        027760  000004                          .WORD   4
        027762  007237                          .WORD   M.CSR2
        027764  015016                          .WORD   ER.REG
   5420
   5421 027766                          400$:   ENDSUB
        027766                          L10076:
        027766  104403                          TRAP    C$ESUB
```

```
5423 027770                            STARS
                                       ;;*****************************************************************
5424                                   ;;
5425                                   ;;    SUBTEST 2:
5426                                   ;;
5427                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
5428                                   ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERROR
5429                                   ;;      ISSUE INTERFACE READ CMD AND SET CYCLE REQUEST BIT
5430                                   ;;      DELAY A FEW CYCLES AND SET IS REGISTER XABT BIT
5431                                   ;;      READ REGISTERS, CHECK IRY BIT SET
5432                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
5433 027770                            STARS
                                       ;;*****************************************************************
5434
5435 027770                                    BGNSUB
     027770                            T13.2:
     027770  104402                            TRAP    C$BSUB
5436
5437 027772  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
5438 027776  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
5439 030002  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
5440
5441 030004                                    ERRDF   4,M.MBC1,ER.DMP         ;DEVICE FATAL
     030004  104455                            TRAP    C$ERDF
     030006  000004                            .WORD   4
     030010  011650                            .WORD   M.MBC1
     030012  015074                            .WORD   ER.DMP
5442 030014                                    DOCLN                           ;CLEAN UP AND EXIT
     030014  104444                            TRAP    C$DCLN
5443
5444 030016  012700  177700            100$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
5445 030022  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
5446 030026  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
5447 030032  005737  003030                    TST     MBCFLG                  ;TEST FOR ERROR
5448 030036  001405                            BEQ     150$                    ;BRANCH IF NO ERROR
5449
5450 030040                                    ERRHRD  5,M.WRD2,ER.DMP         ;"XFER ERROR ON INTERFACE WRITE"
     030040  104456                            TRAP    C$ERHRD
     030042  000005                            .WORD   5
     030044  007324                            .WORD   M.WRD2
     030046  015074                            .WORD   ER.DMP
5451 030050  000463                            BR      400$                    ;EXIT SUBTEST
```

```
5453
5454 030052  012700  177600          150$:  MOV     #-200,R0              ;LOAD BLOCK SIZE = 128 WORDS
5455 030056  004737  015726                 JSR     PC,RDINI              ;AND INITIALIZE ADDRESS REGISTERS
5456 030062  012777  000171  152206         MOV     #RIE.CMD,@A.CS1       ;ISSUE READ COMMAND TO CS1
5457 030070  004737  016030                 JSR     PC,SILODLY            ;WAIT FOR SILO TO FILL
5458 030074  012777  000020  152206         MOV     #IS.CYC,@A.IS         ;SET IS REGISTER CYC BIT
5459 030102  004737  016016                 JSR     PC,DMADLY             ;DELAY A FEW CYCLES
5460
5461 030106  012777  004000  152174         MOV     #IS.XABT,@A.IS        ;SET IS REGISTER TRANSFER ABORT
5462 030114  004737  015444                 JSR     PC,TSTINT             ;WAIT FOR INTERRUPT OR TIME-OUT
5463 030120  004737  015606                 JSR     PC,RDREG              ;READ REGISTERS
5464
5465 030124  032737  000200  002340         BIT     #IS.IRY,R.IS          ;TEST FOR IRY BIT SET
5466 030132  001005                        BNE     200$                  ;YES:  CONTINUE
5467
5468 030134                                ERRHRD  6,M.IRY5,ER.DMP       ;''IRY NOT SET BY WORD TRANSFER ABORT''
     030134  104456                        TRAP    C$ERHRD
     030136  000006                        .WORD   6
     030140  011320                        .WORD   M.IRY5
     030142  015074                        .WORD   ER.DMP
5469 030144  000425                        BR      400$                  ;EXIT SUBTEST
5470
5471 030146  005737  002330          200$:  TST     R.WC                  ;TEST WORD COUNT
5472 030152  100404                        BMI     300$                  ;IF LESS THAN ZERO THEN OK
5473
5474 030154                                ERRHRD  7,M.ABT2,ER.DMP       ;''NO XFER ABORT ON INTERFACE READ''
     030154  104456                        TRAP    C$ERHRD
     030156  000007                        .WORD   7
     030160  006447                        .WORD   M.ABT2
     030162  015074                        .WORD   ER.DMP
5475
5476 030164  012737  144270  002734  300$:  MOV     #144270,FP.EXP        ;EXPECTED VALUE:  SC,TRE,DVA,RDY,READ
5477 030172  023737  002734  002326         CMP     FP.EXP,R.CS1          ;COMPARE CS1 WITH EXPECTED STATUS
5478 030200  001407                        BEQ     400$                  ;IF EQ THEN CONTINUE
5479
5480 030202  013737  002326  002736         MOV     R.CS1,FP.ACT          ;LOAD ACTUAL VALUE
5481 030210                                ERRHRD  8,M.CSR2,ER.REG       ;''CS1 REGISTER STATUS ERROR''
     030210  104456                        TRAP    C$ERHRD
     030212  000010                        .WORD   8
     030214  007237                        .WORD   M.CSR2
     030216  015016                        .WORD   ER.REG
5482
5483
5484 030220                        400$:  ENDSUB
     030220                        L10077:
     030220  104403                        TRAP    C$ESUB
5485
5486 030222  004737  015324                 JSR     PC,CLRISR             ;CLEAR ISR VECTOR
5487
5488 030226  004737  015550                 JSR     PC,CLRTST             ;AND CLEAR TEST DATA
5489
5490 030232                                ENDTST
     030232                        L10075:
     030232  104401                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 14:  CHECK WORD TRANSFER ABORT LOGIC (AT03)
030234                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK WORD MODE DATA TRANSFER ABORT LOGIC THROUGH LOOP-BACK
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT REVA & NOT BYTE MODE & AT3 ABORT NOT DISABLED THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER CLEAR ERROR
                                ;;      NO WRITE/READ TRANSFER ABORT
                                ;;      TRE NOT SET:  ADJUST BANDWIDTH
                                ;;      INTERFACE IRY BIT NOT SET BY TRANSFER ABORT
030234                          STARS
                                ;;*****************************************************************

                                        BGNTST
030234                          T14::
030234

                        000214          SELMASK =       B.REVA + B.BYTE + B.ABORT

030234  013700  003010                  MOV     SELECT,R0               ;READ SELECT WORD
030240  032700  000214                  BIT     #SELMASK,R0             ;TEST REVA,BYTE OR ABORT BIT SET
030244  001402                          BEQ     50$                     ;BRANCH IF ALL BITS RESET

030246                                  EXIT    TST                     ;EXIT TEST
030246  104432                          TRAP    C$EXIT
030250  000476                          .WORD   L10100-.

030252  004737  015274          50$:    JSR     PC,SETISR               ;SETUP ISR VECTOR
030256  013737  002406  002730          MOV     NA.CS1,FP.NAM           ;LOAD REGISTER NAME ADDRESS
030264  013737  002436  002746          MOV     BA.CS1,FP.TBL           ;LOAD REGISTER BIT TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST
```

```
5528 030272                            STARS
                                       ;;*****************************************************************
5529                                   ;;
5530                                   ;;    SUBTEST 1:
5531                                   ;;
5532                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
5533                                   ;;      ISSUE INTERFACE WRITE CMD AND SET CYCLE REQUEST BIT
5534                                   ;;      DELAY A FEW CYCLES AND SET FS REGISTER FUNCTION BIT F3
5535                                   ;;      READ REGISTERS, CHECK IRY BIT SET
5536                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
5537 030272                            STARS
                                       ;;*****************************************************************
5538
5539 030272                                    BGNSUB
     030272                            T14.1:
     030272  104402                           TRAP    C$BSUB
5540
5541 030274  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
5542 030300  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
5543 030304  001405                           BEQ     100$                    ;CONTINUE IF NO ERROR
5544
5545 030306                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
     030306  104455                           TRAP    C$ERDF
     030310  000001                           .WORD   1
     030312  011650                           .WORD   M.MBC1
     030314  015074                           .WORD   ER.DMP
5546 030316                                    DOCLN                           ;CLEAN UP AND EXIT
     030316  104444                           TRAP    C$DCLN
5547
5548 030320  012700  177600          100$:   MOV     #-200,R0                ;LOAD BLOCK SIZE = 128 WORDS
5549 030324  004737  015762                   JSR     PC,WRTINI               ;AND INITIALIZE ADDRESS REGISTERS
5550 030330  012777  000161  151740          MOV     #WIE.CMD,@A.CS1         ;ISSUE WRITE COMMAND TO CS1
5551 030336  004737  016030                   JSR     PC,SILODLY              ;WAIT FOR SILO TO FILL
5552 030342  012777  000020  151740          MOV     #IS.CYC,@A.IS           ;SET IS REGISTER CYC BIT
5553 030350  004737  016016                   JSR     PC,DMADLY               ;DELAY A FEW CYCLES
5554
5555 030354  012777  004000  151722          MOV     #FS.F3,@A.FS            ;SET FS REGISTER FUNCTION BIT F3
5556 030362  004737  015444                   JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME-OUT
5557 030366  004737  015606                   JSR     PC,RDREG                ;READ REGISTERS
5558 030372  012777  000000  151704          MOV     #0,@A.FS                ;CLEAR FUNCTION / STATUS BITS
5559
5560 030400  032737  000200  002340          BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
5561 030406  001005                           BNE     200$                    ;YES:  CONTINUE
5562
5563 030410                                    ERRHRD  2,M.IRY5,ER.DMP         ;"IRY NOT SET BY WORD TRANSFER ABORT"
     030410  104456                           TRAP    C$ERHRD
     030412  000002                           .WORD   2
     030414  011320                           .WORD   M.IRY5
     030416  015074                           .WORD   ER.DMP
5564 030420  000425                           BR      400$                    ;EXIT SUBTEST
5565
5566 030422  005737  002330          200$:   TST     R.WC                    ;TEST WORD COUNT
5567 030426  100404                           BMI     300$                    ;IF LESS THAN ZERO THEN OK
5568
5569 030430                                    ERRHRD  3,M.ABT1,ER.DMP         ;'NO XFER ABORT ON INTERFACE WRITE"
     030430  104456                           TRAP    C$ERHRD
     030432  000003                           .WORD   3
```

```
ZDRMA0 DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 158-1
TEST 14:  CHECK WORD TRANSFER ABORT LOGIC (AT03)

        030434  006375                          .WORD   M.ABT1
        030436  015074                          .WORD   ER.DMP
   5570
   5571 030440  012737  144260  002734  300$:   MOV     #144260,FP.EXP          ;EXPECTED VALUE:  SC,TRE,DVA,RDY,WRITE
   5572 030446  023737  002734  002326          CMP     FP.EXP,R.CS1            ;COMPARE CS1 WITH EXPECTED STATUS
   5573 030454  001407                          BEQ     400$                    ;IF EQ THEN CONTINUE
   5574
   5575 030456  013737  002326  002736          MOV     R.CS1,FP.ACT            ;LOAD ACTUAL VALUE
   5576 030464                                   ERRHRD  4,M.CSR2,ER.REG         ;''CS1 REGISTER STATUS ERROR''
        030464  104456                          TRAP    C$ERHRD
        030466  000004                          .WORD   4
        030470  007237                          .WORD   M.CSR2
        030472  015016                          .WORD   ER.REG
   5577
   5578 030474                          400$:   ENDSUB
        030474                          L10101:
        030474  104403                          TRAP    C$ESUB
```

```
5580 030476                            STARS
                                       ;;*****************************************************************
5581                                   ;;
5582                                   ;;    SUBTEST 2:
5583                                   ;;
5584                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
5585                                   ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERRORS
5586                                   ;;      ISSUE INTERFACE WRITE CMD AND SET CYCLE REQUEST BIT
5587                                   ;;      DELAY A FEW CYCLES AND SET FS REGISTER FUNCTION BIT F5
5588                                   ;;      READ REGISTERS, CHECK IRY BIT SET
5589                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
5590 030476                            STARS
                                       ;;*****************************************************************
5591
5592 030476                                    BGNSUB
     030476                            T14.2:
     030476  104402                            TRAP    C$BSUB
5593
5594 030500  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
5595 030504  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
5596 030510  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
5597
5598 030512                                    ERRDF   4,M.MBC1,ER.DMP         ;DEVICE FATAL
     030512  104455                            TRAP    C$ERDF
     030514  000004                            .WORD   4
     030516  011650                            .WORD   M.MBC1
     030520  015074                            .WORD   ER.DMP
5599 030522                                    DOCLN                           ;CLEAN UP AND EXIT
     030522  104444                            TRAP    C$DCLN
5600
5601 030524  012700  177700            100$:   MOV     #-100,R0                ;LOAD BLOCK SIZE = 64 WORDS
5602 030530  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
5603 030534  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
5604 030540  005737  003030                    TST     MBCFLG                  ;TEST FOR ERROR
5605 030544  001405                            BEQ     150$                    ;BRANCH IF NO ERROR
5606
5607 030546                                    ERRHRD  5,M.WRD2,ER.DMP         ;"XFER ERROR ON INTERFACE WRITE"
     030546  104456                            TRAP    C$ERHRD
     030550  000005                            .WORD   5
     030552  007324                            .WORD   M.WRD2
     030554  015074                            .WORD   ER.DMP
5608 030556  000466                            BR      400$                    ;EXIT SUBTEST
```

```
5610
5611 030560  012700  177600          150$:  MOV     #-200,R0              ;LOAD BLOCK SIZE = 128 WORDS
5612 030564  004737  015726                 JSR     PC,RDINI              ;AND INITIALIZE ADDRESS REGISTERS
5613 030570  012777  000171  151500         MOV     #RIE.CMD,@A.CS1       ;ISSUE READ COMMAND TO CS1
5614 030576  004737  016030                 JSR     PC,SILODLY            ;WAIT FOR SILO TO FILL
5615 030602  012777  000020  151500         MOV     #IS.CYC,@A.IS         ;SET IS REGISTER CYC BIT
5616 030610  004737  016016                 JSR     PC,DMADLY             ;DELAY A FEW CYCLES
5617
5618 030614  012777  004000  151462         MOV     #FS.F3,@A.FS          ;SET FS REGISTER FUNCTION BIT F3
5619 030622  004737  015444                 JSR     PC,TSTINT             ;WAIT FOR INTERRUPT OR TIME-OUT
5620 030626  004737  015606                 JSR     PC,RDREG              ;READ REGISTERS
5621 030632  012777  000000  151444         MOV     #0,@A.FS              ;ZERO ALL FUNCTION BITS
5622
5623 030640  032737  000200  002340         BIT     #IS.IRY,R.IS          ;TEST FOR IRY BIT SET
5624 030646  001005                        BNE     200$                  ;YES:  CONTINUE
5625
5626 030650                                ERRHRD  6,M.IRY5,ER.DMP       ;''IRY NOT SET BY WORD TRANSFER ABORT''
     030650  104456                        TRAP    C$ERHRD
     030652  000006                        .WORD   6
     030654  011320                        .WORD   M.IRY5
     030656  015074                        .WORD   ER.DMP
5627 030660  000425                        BR      400$                  ;EXIT SUBTEST
5628
5629 030662  005737  002330          200$:  TST     R.WC                  ;TEST WORD COUNT
5630 030666  100404                        BMI     300$                  ;IF LESS THAN ZERO THEN OK
5631
5632 030670                                ERRHRD  7,M.ABT2,ER.DMP       ;'NO XFER ABORT ON INTERFACE READ''
     030670  104456                        TRAP    C$ERHRD
     030672  000007                        .WORD   7
     030674  006447                        .WORD   M.ABT2
     030676  015074                        .WORD   ER.DMP
5633
5634 030700  012737  144270  002734  300$:  MOV     #144270,FP.EXP        ;EXPECTED VALUE:  SC,TRE,DVA,RDY,READ
5635 030706  023737  002734  002326         CMP     FP.EXP,R.CS1          ;COMPARE CS1 WITH EXPECTED STATUS
5636 030714  001407                        BEQ     400$                  ;IF EQ THEN CONTINUE
5637
5638 030716  013737  002326  002736         MOV     R.CS1,FP.ACT          ;LOAD ACTUAL VALUE
5639 030724                                ERRHRD  8,M.CSR2,ER.REC       ;''CS1 REGISTER STATUS ERROR''
     030724  104456                        TRAP    C$ERHRD
     030726  000010                        .WORD   8
     030730  007237                        .WORD   M.CSR2
     030732  015016                        .WORD   ER.REG
5640
5641 030734                        400$:  ENDSUB
     030734                        L10102:
     030734  104403                        TRAP    C$ESUB
5642
5643 030736  004737  015324                 JSR     PC,CLRISR             ;CLEAR VECTOR
5644
5645 030742  004737  015550                 JSR     PC,CLRTST             ;AND CLEAR TEST DATA
5646
5647 030746                                ENDTST
     030746                        L10100:
     030746  104401                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 15:  CHECK BYTE TRANSFER LOGIC
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK IRY AND CYC BITS FOR PROPER FUNCTION
                                ;;      AROUND DATA TRANSFER COMMANDS
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF BYTE MODE THEN:
                                ;;          SETUP INTERRUUPT SERVICE VECTOR
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR STATUS
                                ;;          EXECUTE SUBTEST 1-3 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      INTERFACE STATUS IRY BIT NOT SET/RESET
                                ;;      MASSBUS CONTROLLER RDY BIT NOT SET/RESET
                                STARS
                                ;;*****************************************************************
                                        BGNTST
                                T15::

                        000010  SELTEST =       B.BYTE

030750  013700  003010        50$:    MOV     SELECT,R0               ;LOAD TEST SELECT WORD
030754  032700  000010                BIT     #SELTEST,R0             ;TEST BYTE MODE
030760  001002                        BNE     75$                     ;CONTINUE IF BYTE MODE

030762                                EXIT    TST                     ;NO:  EXIT TEST
030762  104432                        TRAP    C$EXIT
030764  000302                        .WORD   L10103-.

030766  004737  015342        75$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
030772  005737  003030                TST     MBCFLG                  ;TEST FLAG FOR ERROR
030776  001405                        BEQ     100$                    ;CONTINUE IF NO ERROR

031000                                ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
031000  104455                        TRAP    C$ERDF
031002  000001                        .WORD   1
031004  011650                        .WORD   M.MBC1
031006  015074                        .WORD   ER.DMP
031010                                DOCLN                           ;CLEAN UP AND EXIT
031010  104444                        TRAP    C$DCLN

031012  004737  015274        100$:   JSR     PC,SETISR               ;SETUP SERVICE ROUTINE VECTOR
                                                                      ;EXECUTE SUBTEST
```

```
031016                                 STARS
                                       ;;*******************************************************************
                                       ;;
                                       ;;     SUBTEST 1:
                                       ;;
                                       ;;       SET CYCLE REQUEST AND DELAY A FEW CYCLES
                                       ;;       READ REGISTERS, CHECK IRY BIT SET AND RDY BIT SET
031016                                 STARS
                                       ;;*******************************************************************
031016                                         BGNSUB                          ;CHECK INTERFACE CLEAR CMD
031016                                 T15.1:
031016  104402                                 TRAP    C$BSUB

031020  012777  000020  151262         100$:   MOV     #IS.CYC,@A.IS           ;ISSUE CYCLE REQUEST
031026  004737  016042                         JSR     PC,DLY                  ;DELAY A FEW CYCLES
031032  004737  015606                         JSR     PC,RDREG                ;READ REGISTERS

031036  032737  000200  002340                 BIT     #IS.IRY,R.IS            ;TEST FOR IRY SET
031044  001005                                 BNE     200$                    ;IRY SET:  CONTINUE

031046                                         ERRHRD  2,M.IRY2,ER.DMP         ;"IRY BIT RESET WHEN CYC BIT SET"
031046  104456                                 TRAP    C$ERHRD
031050  000002                                 .WORD   2
031052  011102                                 .WORD   M.IRY2
031054  015074                                 .WORD   ER.DMP
031056  000410                                 BR      400$                    ;EXIT FROM SUB TEST

031060  032737  000200  002326         200$:   BIT     #CS1.RDY,R.CS1          ;TEST FOR RDY BIT SET
031066  001004                                 BNE     400$                    ;RDY SET:  CONTINUE

031070                                         ERRHRD  3,M.RDY2,ER.DMP         ;"MBC RDY BIT NOT SET BY IRY"
031070  104456                                 TRAP    C$ERHRD
031072  000003                                 .WORD   3
031074  012024                                 .WORD   M.RDY2
031076  015074                                 .WORD   ER.DMP

031100                                 400$:   ENDSUB
031100                                 L10104:
031100  104403                                 TRAP    C$ESUB
```

```
 5716 031102                            STARS
                                        ;;**************************************************************
 5717                                   ;;
 5718                                   ;;    SUBTEST 2:
 5719                                   ;;
 5720                                   ;;      INITIALIZE REGISTERS AND ISSUE INTERFACE WRITE
 5721                                   ;;      READ REGISTERS, CHECK IRY RESET AND RDY RESET
 5722 031102                            STARS
                                        ;;**************************************************************
 5723 031102                            BGNSUB
      031102                            T15.2:
      031102  104402                   TRAP    C$BSUB
 5724
 5725 031104  012700  177770  100$:     MOV     #-10,R0                 ;LOAD BLOCK SIZE = 8 BYTES
 5726 031110  004737  015762           JSR     PC,WRTINI               ;AND SETUP ADDRESS REGISTERS
 5727 031114  012777  000061  151154   MOV     #WR.CMD,@A.CS1          ;ISSUE READ TO INTERFACE
 5728
 5729 031122  004737  015606           JSR     PC,RDREG                ;READ REGISTERS INTO R.TBL
 5730 031126  032737  000200  002340   BIT     #IS.IRY,R.IS            ;AND TEST FOR IRY RESET
 5731 031134  001405                   BEQ     200$                    ;IRY RESET:  CONTINUE
 5732
 5733 031136                           ERRHRD  4,M.IRY3,ER.DMP         ;"IRY NOT RESET BY DATA TRANSFER CMD"
      031136  104456                   TRAP    C$ERHRD
      031140  000004                   .WORD   4
      031142  011162                   .WORD   M.IRY3
      031144  015074                   .WORD   ER.DMP
 5734 031146  000410                   BR      400$                    ;EXIT FROM SUB TEST
 5735
 5736 031150  032737  000200  002326  200$:   BIT     #CS1.RDY,R.CS1          ;TEST FOR RDY RESET
 5737 031156  001404                   BEQ     400$                    ;RDY RESET:  CONTINUE
 5738
 5739 031160                           ERRHRD  5,M.RDY1,ER.DMP         ;"MSC RDY NOT RESET BY IRY"
      031160  104456                   TRAP    C$ERHRD
      031162  000005                   .WORD   5
      031164  011750                   .WORD   M.RDY1
      031166  015074                   .WORD   ER.DMP
 5740
 5741 031170                    400$:   ENDSUB
      031170                    L10105:
      031170  104403                   TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 3:
                                        ;;
                                        ;;      SET CYCLE REQUEST AND TEST FOR INTERRUPT OR TIME OUT
                                        ;;      READ REGISTERS, CHECK IRY BIT SET AND RDY BIT SET
                                        STARS
                                        ;;*****************************************************************
                                        BGNSUB
                                        T15.3:
                                        TRAP    C$BSUB

                                100$:   MOV     #IS.CYC,@A.IS           ;ISSUE CYCLE REQUEST AND WAIT
                                        JSR     PC,TSTINT               ;TEST FOR INTERRUPT OR TIME OUT

                                        JSR     PC,RDREG                ;READ REGISTERS
                                        BIT     #IS.IRY,R.IS            ;TEST FOR IRY SET
                                        BNE     200$                    ;IRY SET:  CONTINUE

                                        ERRHRD  6,M.IRY7,ER.DMP         ;"IRY NOT SET BY BYTE XFER EBL"
                                        TRAP    C$ERHRD
                                        .WORD   6
                                        .WORD   M.IRY7
                                        .WORD   ER.DMP
                                        BR      400$                    ;EXIT SUB TEST

                                200$:   BIT     #IS.IRY,R.CS1           ;TEST FOR RDY SET
                                        BNE     400$                    ;RDY SET:  CONTINUE

                                        ERRHRD  7,M.RDY2,ER.DMP         ;'MBC RDY BIT NOT SET BY IRY"
                                        TRAP    C$ERHRD
                                        .WORD   7
                                        .WORD   M.RDY2
                                        .WORD   ER.DMP

                                400$:   ENDSUB
                                L10106:
                                        TRAP    C$ESUB

                                        JSR     PC,CLRISR               ;CLEAR VECTOR

                                        JSR     PC,CLRTST               ;AND CLEAR TEST DATA

                                        ENDTST
                                L10103:
                                        TRAP    C$ETST
```

```
                                        .SBTTL  TEST 16:  CHECK INTERRUPT ON BYTE TRANSFER
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  TEST DESCRIPTION:
                                        ;;
                                        ;;      CHECK THAT INTERRUPT WAS RECEIVED ON DATA TRANSFER COMPLETION
                                        ;;
                                        ;;    TEST STEPS:
                                        ;;
                                        ;;      IF BYTE MODE THEN:
                                        ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR AND INIT BLOCK SIZE
                                        ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                        ;;
                                        ;;  ERROR CONDITIONS:
                                        ;;
                                        ;;      MASSBUS CONTROLLER CLEAR ERROR
                                        ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                        STARS
                                        ;;*****************************************************************

                                                BGNTST
                                        T16::

                    000010              SELTEST =       B.BYTE

031270  013700  003010                  50$:    MOV     SELECT,R0               ;LOAD TEST SELECT WORD
03 274  032700  000010                          BIT     #SELTEST,R0             ;TEST BYTE MODE
031300  001002                                  BNE     75$                     ;CONTINUE IF BYTE MODE

031302                                          EXIT    TST                     ;NO:  EXIT TEST
031302  104432                                  TRAP    C$EXIT
031304  000242                                  .WORD   L10107-.

031306  004737  015342                  75$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
031312  005737  003030                          TST     MBCFLG                  ;TEST FLAG FOR ERROR
031316  001405                                  BEQ     100$                    ;CONTINUE IF NO ERROR

031320                                          ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
031320  104455                                  TRAP    C$ERDF
031322  000001                                  .WORD   1
031324  011650                                  .WORD   M.MBC1
031326  015074                                  .WORD   ER.DMP
031330                                          DOCLN                           ;CLEAN UP AND EXIT
031330  104444                                  TRAP    C$DCLN

031332  004737  015274                  100$:   JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
```

```
 5815 031336                                    STARS
                                                ;;*****************************************************************
 5816                                           ;;
 5817                                           ;;    SUBTEST 1:
 5818                                           ;;
 5819                                           ;;      WRITE BLOCK TO INTERFACE
 5820                                           ;;      CHECK FOR INTERRUPT RECEIVED
 5821 031336                                    STARS
                                                ;;*****************************************************************
 5822 031336                                            BGNSUB
      031336                                    T16.1:
      031336  104402                                   TRAP    C$BSUB
 5823
 5824 031340  012700  177770                       MOV     #-10,R0                 ;LOAD BLOCK SIZE = 8 BYTES
 5825 031344  004737  015674                       JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 5826 031350  004737  015410                       JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
 5827 031354  005737  003030                       TST     MBCFLG                  ;TEST FOR ERROR
 5828 031360  001405                               BEQ     150$                    ;BRANCH IF NO ERROR
 5829
 5830 031362                                       ERRHRD  2,M.BYT2,ER.DMP         ;'BYTE XFER ERROR ON INTERFACE WRITE"
      031362  104456                              TRAP    C$ERHRD
      031364  000002                              .WORD   2
      031366  007533                              .WORD   M.BYT2
      031370  015074                              .WORD   ER.DMP
 5831 031372  000420                               BR      400$                    ;EXIT SUBTEST
 5832
 5833 031374  032737  000200  002340  150$:      BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
 5834 031402  001005                               BNE     200$                    ;IRY SET:  CONTINUE
 5835
 5836 031404                                       ERRHRD  3,M.BYT1,ER.DMP         ;'DATA TRANSFER NOT COMPLETE"
      031404  104456                              TRAP    C$ERHRD
      031406  000003                              .WORD   3
      031410  007500                              .WORD   M.BYT1
      031412  015074                              .WORD   ER.DMP
 5837 031414  000407                               BR      400$                    ;EXIT SUB TEST
 5838
 5839 031416  005737  003022            200$:      TST     INTFLG                  ;TEST FOR INTERRUPT RECEIVED
 5840 031422  001004                               BNE     400$                    ;BRANCH IF NOT ZERO
 5841
 5842 031424                                       ERRHRD  4,M.INT2,ER.DMP         ;'NO INTERRUPT ON BYTE XFER EBL"
      031424  104456                              TRAP    C$ERHRD
      031426  000004                              .WORD   4
      031430  010761                              .WORD   M.INT2
      031432  015074                              .WORD   ER.DMP
 5843
 5844 031434                            400$:      ENDSUB                          ;WRITE BLOCK
      031434                            L10110:
      031434  104403                              TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      READ BLOCK FROM INTERFACE
                                        ;;      CHECK FOR INTERRUPT RECEIVED
                                        STARS
                                        ;;*****************************************************************
                                        BGNSUB
                                        T16.2:
                                        TRAP    C$BSUB

                                        MOV     #-10,R0                 ;LOAD BLOCK SIZE = 8 BYTES
                                        JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
                                        JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
                                        TST     MBCFLG                  ;TEST FOR ERROR
                                        BEQ     150$                    ;BRANCH IF NO ERROR

                                        ERRHRD  5,M.BYT3,ER.DMP         ;'BYTE XFER ERROR ON INTERFACE READ''
                                        TRAP    C$ERHRD
                                        .WORD   5
                                        .WORD   M.BYT3
                                        .WORD   ER.DMP
                                        BR      400$                    ;EXIT SUBTEST

                                150$:   BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
                                        BNE     200$                    ;IRY SET:  CONTINUE

                                        ERRHRD  6,M.BYT1,ER.DMP         ;'BYTE TRANSFER NOT COMPLETE''
                                        TRAP    C$ERHRD
                                        .WORD   6
                                        .WORD   M.BYT1
                                        .WORD   ER.DMP
                                        BR      400$                    ;EXIT SUB TEST

                                200$:   TST     INTFLG                  ;TEST FOR INTERRUPT RECEIVED
                                        BNE     400$                    ;BRANCH IF NOT ZERO

                                        ERRHRD  7,M.INT2,ER.DMP         ;'NO INTERRUPT ON BYTE XFER EBL''
                                        TRAP    C$ERHRD
                                        .WORD   7
                                        .WORD   M.INT2
                                        .WORD   ER.DMP

                                400$:   ENDSUB                          ;READ BLOCK
                                L10111:
                                        TRAP    C$ESUB

                                        JSR     PC,CLRISR               ;AND RESTORE VECTOR ADDRESS

                                        JSR     PC,CLRTST               ;AND CLEAR TEST DATA

                                        ENDTST
                                L10107:
                                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 17:  CHECK BYTE BLOCK DATA TRANSFER
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      PERFORM 64-BYTE DMA DATA TRANSFERS,
                                ;;      COMPARE DATA READ AGAINST DATA WRITTEN
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF BYTE MODE THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          INITIALIZE TRANSFER PARAMETERS AND
                                ;;          EXECUTE SUBTEST 1-4 AND CLEAR INTERRUPT VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      DATA TRANSFER NOT COMPLETE
                                ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                ;;      DATA RECEIVED DOES NOT MATCH EXPECTED DATA
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T17::

                        000010  SELTEST =       B.BYTE

031550  013700  003010                MOV     SELECT,R0               ;READ SELECT WORD
031554  032700  000010                BIT     #SELTEST,R0             ;TEST IF BYTE SELECT
031560  001002                      BNE     50$                     ;BRANCH IF SET

031562                              EXIT    TST                     ;NOT BYTE MODE:  EXIT TEST
031562  104432                      TRAP    C$EXIT
031564  001756                      .WORD   L10112-.

031566  004737  015274              50$:    JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
031572  012737  000100  002512              MOV     #100,D.MAX              ;INITIALIZE MAX DATA BUFFER INDEX
031600  012737  000001  002510              MOV     #1,D.LPCNT              ;INITIALIZE TRANSFER LOOP COUNTER

031606                          LP.BXFR:                                ;TRANSFER LOOP LABEL
```

```
 5923 031606                            STARS
                                        ;;*****************************************************************
 5924                                   ;;
 5925                                   ;;    SUBTEST 1:
 5926                                   ;;
 5927                                   ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 5928                                   ;;      SETUP DATA BUFFER FOR ALL ONE'S
 5929                                   ;;
 5930                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 5931                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 5932                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 5933                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5934                                   ;;
 5935                                   ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 5936                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 5937                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 5938                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 5939                                   ;;
 5940                                   ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 5941 031606                            STARS
                                        ;;*****************************************************************
 5942
 5943 031606                                    BGNSUB
      031606                            T17.1:
      031606  104402                           TRAP    C$BSUB
 5944
 5945 031610  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 5946 031614  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 5947 031620  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 5948
 5949 031622                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      031622  104455                           TRAP    C$ERDF
      031624  000001                           .WORD   1
      031626  011650                           .WORD   M.MBC1
      031630  015074                           .WORD   ER.DMP
 5950 031632                                    DOCLN                           ;CLEAN UP AND EXIT
      031632  104444                           TRAP    C$DCLN
 5951
 5952 031634  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 5953
 5954 031636  012762  177777  004054    150$:   MOV     #177777,OBUF(R2)        ;LOAD ONE'S INTO BUFFER
 5955 031644  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 5956 031652  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 5957 031656  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 5958 031662  003365                            BGT     150$                    ;CONTINUE UNTIL BUFFER INITIALIZED
 5959
 5960 031664  012700  177740            200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 5961 031670  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 5962 031674  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 5963 031700  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
 5964 031704  001005                            BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 5965
 5966 031706                                    ERRHRD  2,M.INT2,ER.DMP         ;"NO INTERRUPT ON BYTE XFER EBL"
      031706  104456                           TRAP    C$ERHRD
      031710  000002                           .WORD   2
      031712  010761                           .WORD   M.INT2
      031714  015074                           .WORD   ER.DMP
```

```
 5967 031716  000407                          BR      300$                    ;EXIT SEGMENT
 5968
 5969 031720  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
 5970 031724  001404                          BEQ     300$                    ;CONTINUE IF RESET
 5971
 5972 031726                                   ERRHRD  3,M.BYT2,ER.DMP         ;'BYTE XFER ERROR ON INTERFACE WRITE''
      031726  104456                          TRAP    C$ERHRD
      031730  000003                          .WORD   3
      031732  007533                          .WORD   M.BYT2
      031734  015074                          .WORD   ER.DMP
 5973
 5974 031736  012700  177740          300$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 5975 031742  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
 5976 031746  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 5977 031752  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
 5978 031756  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
 5979
 5980 031760                                   ERRHRD  4,M.INT2,ER.DMP         ;'NO INTERRUPT ON BYTE XFER EBL''
      031760  104456                          TRAP    C$ERHRD
      031762  000004                          .WORD   4
      031764  010761                          .WORD   M.INT2
      031766  015074                          .WORD   ER.DMP
 5981 031770  000407                          BR      400$                    ;EXIT SEGMENT
 5982
 5983 031772  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
 5984 031776  001404                          BEQ     400$                    ;CONTINUE IF RESET
 5985
 5986 032000                                   ERRHRD  5,M.BYT3,ER.DMP         ;'BYTE XFER ERROR ON INTERFACE READ''
      032000  104456                          TRAP    C$ERHRD
      032002  000005                          .WORD   5
      032004  007602                          .WORD   M.BYT3
      032006  015074                          .WORD   ER.DMP
```

```
 5988
 5989 032010  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 5990 032012  005037  003026                     CLR     ERRFLG                  ;CLEAR ERROR COUNT
 5991
 5992 032016  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
 5993 032024  001427                            BEQ     475$                    ;IDENTICAL:  CONTINUE
 5994
 5995 032026  005737  003026                     TST     ERRFLG                  ;TEST ERROR COUNT
 5996 032032  001004                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
 5997
 5998 032034                                    ERRHRD  C.M.BYT4,ER.DATA        ;''DATA COMPARISON ERROR''
      032034  104456                            TRAP    C$ERHRD
      032036  000006                            .WORD   6
      032040  007650                            .WORD   M.BYT4
      032042  015172                            .WORD   ER.DATA
 5999
 6000 032044  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
 6001 032050  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
 6002 032056  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
 6003
 6004 032060  010237  002742                     MOV     R2,FP.NDX               ;SAVE DATA INDEX
 6005 032064  016237  003054  002736             MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
 6006 032072  016237  004054  002734             MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
 6007 032100  004737  016374                     JSR     PC,DSPDATA              ;AND DISPLAY DATA
 6008
 6009 032104  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
 6010 032110  023702  002512                     CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6011 032114  003340                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
 6012
 6013 032116  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
 6014 032122  001414                            BEQ     600$                    ;BRANCH IF NO ERRORS
 6015
 6016 032124                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
      032124  013746  003026                     MOV     ERRFLG,-(SP)
      032130  012746  013304                     MOV     #F.ERCNT,-(SP)
      032134  012746  000002                     MOV     #2,-(SP)
      032140  010600                             MOV     SP,R0
      032142  104415                             TRAP    C$PNTX
      032144  062706  000006                     ADD     #6,SP
 6017 032150  005237  003004                     INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
 6018
 6019 032154                            600$:   ENDSUB
      032154                            L10113:
      032154  104403                            TRAP    C$ESUB
```

```
 6021 032156                                    STARS
                                                ;;*****************************************************************
 6022                                           ;;
 6023                                           ;;    SUBTEST 2:
 6024                                           ;;
 6025                                           ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 6026                                           ;;      SETUP DATA BUFFER FOR ALL ZERO'S
 6027                                           ;;
 6028                                           ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 6029                                           ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 6030                                           ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 6031                                           ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 6032                                           ;;
 6033                                           ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 6034                                           ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 6035                                           ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 6036                                           ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 6037                                           ;;
 6038                                           ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 6039 032156                                    STARS
                                                ;;*****************************************************************
 6040
 6041 032156                                    T17.2:  BGNSUB                          ;ALL ZERO'S
      032156
      032156  104402                                    TRAP    C$BSUB
 6042
 6043 032160  004737  015342                            JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6044 032164  005737  003030                            TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6045 032170  001405                                    BEQ     100$                    ;CONTINUE IF NO ERROR
 6046
 6047 032172                                            ERRDF   7,M.MBC1,ER.DMP         ;DEVICE FATAL
      032172  104455                                    TRAP    C$ERDF
      032174  000007                                    .WORD   7
      032176  011650                                    .WORD   M.MBC1
      032200  015074                                    .WORD   ER.DMP
 6048 032202                                            DOCLN                           ;CLEAN UP AND EXIT
      032202  104444                                    TRAP    C$DCLN
 6049
 6050 032204  005002                    100$:   CLR     R2                              ;CLEAR BUFFER INDEX REGISTER
 6051
 6052 032206  012762  000000  004054    150$:   MOV     #0,OBUF(R2)                     ;LOAD ZERO'S INTO BUFFER
 6053 032214  012762  000000  003054            MOV     #0,IBUF(R2)                     ;CLEAR INPUT BUFFER
 6054 032222  062702  000002                    ADD     #2,R2                           ;AND BUMP INDEX
 6055 032226  023702  002512                    CMP     D.MAX,R2                        ;COMPARE WITH MAX BUFFER INDEX
 6056 032232  003365                            BGT     150$                            ;CONTINUE UNTIL BUFFER INITIALIZED
 6057
 6058 032234  012700  177740            200$:   MOV     #-40.R0                         ;LOAD BLOCK SIZE = 64 BYTES
 6059 032240  004737  015674                    JSR     PC,WRTBLK                       ;WRITE BLOCK TO INTERFACE
 6060 032244  004737  015410                    JSR     PC,TSTMBC                       ;READ REGISTERS AND TEST ERRORS
 6061 032250  005737  003022                    TST     INTFLG                          ;TEST FOR INTERRUPT
 6062 032254  001005                            BNE     250$                            ;BRANCH IF INTERRUPT RECEIVED
 6063
 6064 032256                                    ERRHRD  8,M.INT2,ER.DMP         ;"NO INTERRUPT ON BYTE XFER EBL"
      032256  104456                            TRAP    C$ERHRD
      032260  000010                            .WORD   8
      032262  010761                            .WORD   M.INT2
      032264  015074                            .WORD   ER.DMP
```

```
 6065 032266  000407                          BR      300$                    ;EXIT SEGMENT
 6066
 6067 032270  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
 6068 032274  001404                          BEQ     300$                    ;CONTINUE IF RESET
 6069
 6070 032276                                   ERRHRD  9,M.BYT2,ER.DMP         ;'BYTE XFER ERROR ON INTERFACE WRITE''
      032276  104456                          TRAP    C$ERHRD
      032300  000011                          .WORD   9
      032302  007533                          .WORD   M.BYT2
      032304  015074                          .WORD   ER.DMP
 6071
 6072 032306  012700  177740          300$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 6073 032312  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
 6074 032316  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 6075 032322  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
 6076 032326  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
 6077
 6078 032330                                   ERRHRD  10,M.INT2,ER.DMP        ;'NO INTERRUPT ON BYTE XFER EBL''
      032330  104456                          TRAP    C$ERHRD
      032332  000012                          .WORD   10
      032334  010761                          .WORD   M.INT2
      032336  015074                          .WORD   ER.DMP
 6079 032340  000407                          BR      400$                    ;EXIT SEGMENT
 6080
 6081 032342  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
 6082 032346  001404                          BEQ     400$                    ;CONTINUE IF RESET
 6083
 6084 032350                                   ERRHRD  11,M.BYT3,ER.DMP        ;'BYTE XFER ERROR ON INTERFACE READ''
      032350  104456                          TRAP    C$ERHRD
      032352  000013                          .WORD   11
      032354  007602                          .WORD   M.BYT3
      032356  015074                          .WORD   ER.DMP
```

```
6086
6087 032360  005002                    400$:  CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
6088 032362  005037  003026                   CLR     ERRFLG                  ;CLEAR ERROR COUNT
6089
6090 032366  026262  003054  004054    425$:  CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
6091 032374  001427                           BEQ     475$                    ;IDENTICAL:  CONTINUE
6092
6093 032376  005737  003026                   TST     ERRFLG                  ;TEST ERROR COUNT
6094 032402  001004                           BNE     450$                    ;BRANCH IF NOT FIRST ERROR
6095
6096 032404                                   ERRHRD  12,M.BYT4,ER.DATA       ;''DATA COMPARISON ERROR''
     032404  104456                           TRAP    C$ERHRD
     032406  000014                           .WORD   12
     032410  007650                           .WORD   M.BYT4
     032412  015172                           .WORD   ER.DATA
6097
6098 032414  005237  003026            450$:  INC     ERRFLG                  ;BUMP ERROR COUNT
6099 032420  022737  000010  003026           CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
6100 032426  002412                           BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
6101
6102 032430  010237  002742                   MOV     R2,FP.NDX               ;SAVE DATA INDEX
6103 032434  016237  003054  002736           MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
6104 032442  016237  004054  002734           MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
6105 032450  004737  016374                   JSR     PC,DSPDATA              ;AND DISPLAY DATA
6106
6107 032454  062702  000002            475$:  ADD     #2,R2                   ;BUMP INDEX
6108 032460  023702  002512                   CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
6109 032464  003340                           BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
6110
6111 032466  005737  003026            500$:  TST     ERRFLG                  ;TEST NUMBER ERRORS
6112 032472  001414                           BEQ     600$                    ;BRANCH IF NO ERRORS
6113
6114 032474                                   PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     032474  013746  003026                   MOV     ERRFLG,-(SP)
     032500  012746  013304                   MOV     #F.ERCNT,-(SP)
     032504  012746  000002                   MOV     #2,-(SP)
     032510  010600                           MOV     SP,R0
     032512  104415                           TRAP    C$PNTX
     032514  062706  000006                   ADD     #6,SP
6115 032520  005237  003004                   INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
6116
6117 032524                            600$:  ENDSUB
     032524                            L10114:
     032524  104403                           TRAP    C$ESUB
```

```
 6119 032526                            STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 3:
                                        ;;
                                        ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      SETUP DATA BUFFER WITH MOVING ONE PATTERN
                                        ;;
                                        ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
                                        ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
                                        ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
                                        ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
                                        ;;
                                        ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
                                        ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
                                        ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
                                        ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
                                        ;;
                                        ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 6137 032526                            STARS
                                        ;;*****************************************************************
 6138
 6139 032526                                    BGNSUB                          ;MOVING ONE PATTERN
      032526                            T17.3:
      032526  104402                            TRAP    C$BSUB
 6140
 6141 032530  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6142 032534  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6143 032540  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6144
 6145 032542                                    ERRDF   13,M.MBC1,ER.DMP        ;DEVICE FATAL
      032542  104455                            TRAP    C$ERDF
      032544  000015                            .WORD   13
      032546  011650                            .WORD   M.MBC1
      032550  015074                            .WORD   ER.DMP
 6146 032552                                    DOCLN                           ;CLEAN UP AND EXIT
      032552  104444                            TRAP    C$DCLN
 6147
 6148 032554  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 6149 032556  012705  000010                    MOV     #8.,R5                  ;INITIALIZE COUNTER
 6150
 6151 032562  012704  000401            125$:   MOV     #401,R4                 ;INITIALIZE PATTERN REGISTER
 6152
 6153 032566  010462  004054            150$:   MOV     R4,OBUF(R2)             ;LOAD ONES INTO BUFFER
 6154 032572  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 6155 032600  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 6156 032604  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6157 032610  001405                            BEQ     200$                    ;BRANCH IF END OF BUFFER
 6158
 6159 032612  006304                            ASL     R4                      ;SHIFT PATTERN
 6160 032614  105704                            TSTB    R4                      ;COMPARE WITH LAST PATTERN
 6161 032616  001363                            BNE     150$                    ;MOVE AGAIN IF NOT SAME
 6162 032620  005305                            DEC     R5                      ;DECREMENT BLOCK COUNTER
 6163 032622  001357                            BNE     125$                    ;AND CONTINUE IF ANOTHER BLOCK
 6164
 6165 032624  012700  177740            200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 6166 032630  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
```

```
6167 032634  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
6168 032640  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
6169 032644  001005                            BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
6170
6171 032646                                    ERRHRD  14,M.INT2,ER.DMP        ;'NO INTERRUPT ON BYTE XFER EBL''
     032646  104456                            TRAP    C$ERHRD
     032650  000016                            .WORD   14
     032652  010761                            .WORD   M.INT2
     032654  015074                            .WORD   ER.DMP
6172 032656  000407                            BR      300$                    ;EXIT SEGMENT
6173
6174 032660  005737  003030          250$:     TST     MBCFLG                  ;TEST FOR ERROR
6175 032664  001404                            BEQ     300$                    ;CONTINUE IF RESET
6176
6177 032666                                    ERRHRD  15,M.BYT2,ER.DMP        ;'BYTE XFER ERROR ON INTERFACE WRITE''
     032666  104456                            TRAP    C$ERHRD
     032670  000017                            .WORD   15
     032672  007533                            .WORD   M.BYT2
     032674  015074                            .WORD   ER.DMP
6178
6179 032676  012700  177740          300$:     MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
6180 032702  004737  015642                    JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
6181 032706  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
6182 032712  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
6183 032716  001005                            BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
6184
6185 032720                                    ERRHRD  16,M.INT2,ER.DMP        ;'NO INTERRUPT ON BYTE XFER EBL''
     032720  104456                            TRAP    C$ERHRD
     032722  000020                            .WORD   16
     032724  010761                            .WORD   M.INT2
     032726  015074                            .WORD   ER.DMP
6186 032730  000407                            BR      400$                    ;EXIT SEGMENT
6187
6188 032732  005737  003030          350$:     TST     MBCFLG                  ;TEST FOR ERROR
6189 032736  001404                            BEQ     400$                    ;CONTINUE IF RESET
6190
6191 032740                                    ERRHRD  17,M.BYT3,ER.DMP        ;'BYTE XFER ERROR ON INTERFACE READ''
     032740  104456                            TRAP    C$ERHRD
     032742  000021                            .WORD   17
     032744  007602                            .WORD   M.BYT3
     032746  015074                            .WORD   ER.DMP
```

```
6193
6194 032750  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
6195 032752  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
6196
6197 032756  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
6198 032764  001427                            BEQ     475$                    ;IDENTICAL:  CONTINUE
6199
6200 032766  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
6201 032772  001004                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
6202
6203 032774                                    ERRHRD  18,M.BYT4,ER.DATA       ;'DATA COMPARISON ERROR''
     032774  104456                            TRAP    C$ERHRD
     032776  000022                            .WORD   18
     033000  007650                            .WORD   M.BYT4
     033002  015172                            .WORD   ER.DATA
6204
6205 033004  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
6206 033010  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
6207 033016  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
6208
6209 033020  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
6210 033024  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
6211 033032  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
6212 033040  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
6213
6214 033044  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
6215 033050  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
6216 033054  003340                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
6217
6218 033056  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
6219 033062  001414                            BEQ     600$                    ;BRANCH IF NO ERRORS
6220
6221 033064                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     033064  013746  003026                    MOV     ERRFLG,-(SP)
     033070  012746  013304                    MOV     #F.ERCNT,-(SP)
     033074  012746  000002                    MOV     #2,-(SP)
     033100  010600                            MOV     SP,R0
     033102  104415                            TRAP    C$PNTX
     033104  062706  000006                    ADD     #6,SP
6222 033110  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
6223
6224 033114                            600$:   ENDSUB
     033114                            L10115:
     033114  104403                            TRAP    C$ESUB
```

```
 6226 033116                            STARS
                                        ;;*****************************************************************
 6227                                   ;;
 6228                                   ;;    SUBTEST 4:
 6229                                   ;;
 6230                                   ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 6231                                   ;;      SETUP DATA BUFFER WITH MOVING ZERO PATTERN
 6232                                   ;;
 6233                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 6234                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 6235                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 6236                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 6237                                   ;;
 6238                                   ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 6239                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 6240                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 6241                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 6242                                   ;;
 6243                                   ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 6244 033116                            STARS
                                        ;;*****************************************************************
 6245
 6246 033116                                    BGNSUB                          ;MOVING ZERO PATTERN
      033116                            T17.4:
      033116  104402                            TRAP    C$BSUB
 6247
 6248 033120  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6249 033124  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6250 033130  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6251
 6252 033132                                    ERRDF   19,M.MBC1,ER.DMP        ;DEVICE FATAL
      033132  104455                            TRAP    C$ERDF
      033134  000023                            .WORD   19
      033136  011650                            .WORD   M.MBC1
      033140  015074                            .WORD   ER.DMP
 6253 033142                                    DOCLN                           ;CLEAN UP AND EXIT
      033142  104444                            TRAP    C$DCLN
 6254
 6255 033144  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 6256 033146  012705  000010                    MOV     #8.,R5                  ;INITIALIZE COUNTER
 6257
 6258 033152  012704  177376            125$:   MOV     #177376,R4              ;INITIALIZE PATTERN REGISTER
 6259
 6260 033156  010462  004054            150$:   MOV     R4,OBUF(R2)             ;LOAD ONES INTO BUFFER
 6261 033162  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 6262 033170  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 6263 033174  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6264 033200  001412                            BEQ     200$                    ;BRANCH IF END OF BUFFER
 6265
 6266 033202  006304                            ASL     R4                      ;SHIFT PATTERN
 6267 033204  052704  000001                    BIS     #BIT00,R4               ;SET INCOMING BIT
 6268 033210  052704  000400                    BIS     #BIT08,R4               ;FOR BOTH BYTES
 6269 033214  022704  177777                    CMP     #-1,R4                  ;TEST FOR ALL BITS ONE
 6270 033220  001356                            BNE     150$                    ;NO:  CONTINUE
 6271
 6272 033222  005305                            DEC     R5                      ;DECREMENT BLOCK COUNTER
 6273 033224  001352                            BNE     125$                    ;AND CONTINUE IF ANOTHER BLOCK
```

```
 6274
 6275 033226  012700  177740          200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 6276 033232  004737  015674                  JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 6277 033236  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 6278 033242  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
 6279 033246  001005                          BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 6280
 6281 033250                                  ERRHRD  20,M.INT2,ER.DMP        ;'NO INTERRUPT ON BYTE XFER EBL''
      033250  104456                         TRAP    C$ERHRD
      033252  000024                         .WORD   20
      033254  010761                         .WORD   M.INT2
      033256  015074                         .WORD   ER.DMP
 6282 033260  000407                         BR      300$                    ;EXIT SEGMENT
 6283
 6284 033262  005737  003030          250$:   TST     MBCFLG                  ;TEST FOR ERROR
 6285 033266  001404                          BEQ     300$                    ;CONTINUE IF RESET
 6286
 6287 033270                                  ERRHRD  21,M.BYT2,ER.DMP        ;'BYTE XFER ERROR ON INTERFACE WRITE''
      033270  104456                         TRAP    C$ERHRD
      033272  000025                         .WORD   21
      033274  007533                         .WORD   M.BYT2
      033276  015074                         .WORD   ER.DMP
 6288
 6289 033300  012700  177740          300$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 6290 033304  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
 6291 033310  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 6292 033314  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
 6293 033320  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
 6294
 6295 033322                                  ERRHRD  22,M.INT2,ER.DMP        ;'NO INTERRUPT ON BYTE XFER EBL''
      033322  104456                         TRAP    C$ERHRD
      033324  000026                         .WORD   22
      033326  010761                         .WORD   M.INT2
      033330  015074                         .WORD   ER.DMP
 6296 033332  000407                         BR      400$                    ;EXIT SEGMENT
 6297
 6298 033334  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
 6299 033340  001404                          BEQ     400$                    ;CONTINUE IF RESET
 6300
 6301 033342                                  ERRHRD  23,M.BYT3,ER.DMP        ;'BYTE XFER ERROR ON INTERFACE READ''
      033342  104456                         TRAP    C$ERHRD
      033344  000027                         .WORD   23
      033346  007602                         .WORD   M.BYT3
      033350  015074                         .WORD   ER.DMP
```

```
 6303
 6304 033352  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 6305 033354  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
 6306
 6307 033360  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
 6308 033366  001427                            BEQ     475$                    ;IDENTICAL:  CONTINUE
 6309
 6310 033370  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
 6311 033374  001004                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
 6312
 6313 033376                                    ERRHRD  24,M.BYT4,FR.DATA       ;''DATA COMPARISON ERROR''
      033376  104456                            TRAP    C$ERHRD
      033400  000030                            .WORD   24
      033402  007650                            .WORD   M.BYT4
      033404  015172                            .WORD   ER.DATA
 6314
 6315 033406  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
 6316 033412  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
 6317 033420  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
 6318
 6319 033422  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
 6320 033426  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
 6321 033434  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
 6322 033442  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
 6323
 6324 033446  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
 6325 033452  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6326 033456  003340                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
 6327
 6328 033460  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
 6329 033464  001414                            BEQ     600$                    ;BRANCH IF NO ERRORS
 6330
 6331 033466                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
      033466  013746  003026                    MOV     ERRFLG,-(SP)
      033472  012746  013304                    MOV     #F.ERCNT,-(SP)
      033476  012746  000002                    MOV     #2,-(SP)
      033502  010600                            MOV     SP,R0
      033504  104415                            TRAP    C$PNTX
      033506  062706  000006                    ADD     #6,SP
 6332 033512  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
 6333
 6334 033516                            600$:   ENDSUB
      033516                            L10116:
      033516  104403                            TRAP    C$ESUB
 6335
 6336 033520  005337  002510                    DEC     D.LPCNT                 ;DECREMENT COUNTER
 6337 033524  001402                            BEQ     800$                    ;AND EXIT IF ZERO
 6338 033526  000137  031606                    JMP     LP.BXFR                 ;CONTINUE TRANSFER
 6339
 6340 033532  004737  015324            800$:   JSR     PC,CLRISR               ;CLEAR INTERRUPT SERVICE VECTOR
 6341
 6342 033536  004737  015550                    JSR     PC,CLRTST               ;AND CLEAR TEST DATA
 6343
 6344 033542                                    ENDTST
      033542                            L10112:
      033542  104401                            TRAP    C$ETST
```

```
                                .SBTTL  TEST 18:  CHECK BYTE TRANSFER ABORT LOGIC (XABT)
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK BYTE DATA TRANSFER ABORT LOGIC THROUGH XABT BIT
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF BYTE MODE & NOT REVA THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER CLEAR ERROR
                                ;;      NO WRITE/READ TRANSFER ABORT
                                ;;      TRE NOT SET:  ADJUST BANDWIDTH
                                ;;      INTERFACE IRY BIT NOT SET BY TRANSFER ABORT
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T18::

                        000027  SELMASK =       B.UPAR + B.ABORT + B.AT3 + B.AT0
                        000010  SELTEST =       B.BYTE

033544  013700  003010          MOV     SELECT,R0               ;READ SELECT WORD
033550  042700  000027          BIC     #SELMASK,R0             ;CLEAR DON T CARE SELECT BITS
033554  022700  000010          CMP     #SELTEST,R0             ;COMPARE WITH BYTE MODE
033560  001402                  BEQ     50$                     ;BRANCH IF EQUAL

033562                  25$:    EXIT    TST                     ;REV "A" OR NOT BYTE MODE:  EXIT TEST
033562  104432                  TRAP    C$EXIT
033564  000462                  .WORD   L10117-.

033566  004737  015274  50$:    JSR     PC,SETISR               ;SETUP ISR VECTOR
033572  013737  002406  002730  MOV     NA.CS1,FP.NAM           ;LOAD REGISTER NAME ADDRESS
033600  013737  002436  002746  MOV     BA.CS1,FP.TBL           ;LOAD REGISTER BIT TABLE ADDRESS
                                                                ;EXECUTE SUBTEST
```

```
 6384 033606                            STARS
                                        ;;*****************************************************************
 6385                                   ;;
 6386                                   ;;    SUBTEST 1:
 6387                                   ;;
 6388                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 6389                                   ;;      ISSUE INTERFACE WRITE CMD AND SET CYCLE REQUEST BIT
 6390                                   ;;      DELAY A FEW CYCLES AND SET IS REGISTER XABT BIT
 6391                                   ;;      READ REGISTERS, CHECK IRY BIT SET
 6392                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
 6393 033606                            STARS
                                        ;;*****************************************************************
 6394
 6395 033606                                    BGNSUB
      033606                            T18.1:
      033606  104402                           TRAP    C$BSUB
 6396
 6397 033610  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6398 033614  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6399 033620  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6400
 6401 033622                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      033622  104455                           TRAP    C$ERDF
      033624  000001                           .WORD   1
      033626  011650                           .WORD   M.MBC1
      033630  015074                           .WORD   ER.DMP
 6402 033632                                    DOCLN                           ;CLEAN UP AND EXIT
      033632  104444                           TRAP    C$DCLN
 6403
 6404 033634  012700  177600            100$:   MOV     #-200,R0                ;LOAD BLOCK SIZE = 256 BYTES
 6405 033640  004737  015762                    JSR     PC,WRTINI               ;AND INITIALIZE ADDRESS REGISTERS
 6406 033644  012777  000161  146424            MOV     #WIE.CMD,@A.CS1         ;ISSUE WRITE COMMAND TO CS1
 6407 033652  004737  016030                    JSR     PC,SILODLY              ;WAIT FOR SILO TO FILL
 6408 033656  012777  000020  146424            MOV     #IS.CYC,@A.IS           ;SET IS REGISTER CYC BIT
 6409 033664  004737  016016                    JSR     PC,DMADLY               ;DELAY A FEW CYCLES
 6410
 6411 033670  012777  004000  146412            MOV     #IS.XABT,@A.IS          ;SET IS REGISTER TRANSFER ABORT
 6412 033676  004737  015444                    JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME-OUT
 6413 033702  004737  015606                    JSR     PC,RDREG                ;READ REGISTERS
 6414
 6415 033706  032737  000200  002340            BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
 6416 033714  001005                            BNE     200$                    ;YES:  CONTINUE
 6417
 6418 033716                                    ERRHRD  2,M.IRY8,ER.DMP         ;"IRY NOT SET BY BYTE TRANSFER ABORT"
      033716  104456                           TRAP    C$ERHRD
      033720  000002                           .WORD   2
      033722  011500                           .WORD   M.IRY8
      033724  015074                           .WORD   ER.DMP
 6419 033726  000425                           BR      400$                    ;EXIT SUBTEST
 6420
 6421 033730  005737  002330            200$:   TST     R.WC                    ;TEST WORD COUNT
 6422 033734  100404                            BMI     300$                    ;IF LESS THAN ZERO THEN OK
 6423
 6424 033736                                    ERRHRD  3,M.ABT3,ER.DMP         ;"NO XFER ABORT ON INTERFACE WRITE"
      033736  104456                           TRAP    C$ERHRD
      033740  000003                           .WORD   3
      033742  006520                           .WORD   M.ABT3
```

```
        033744  015074                          .WORD   ER.DMP
  6425
  6426 033746  012737  144260  002734  300$:   MOV     #144260,FP.EXP          ;EXPECTED VALUE:  SC,TRE,DVA,RDY,WRITE
  6427 033754  023737  002734  002326          CMP     FP.EXP,R.CS1            ;COMPARE CS1 WITH EXPECTED STATUS
  6428 033762  001407                          BEQ     400$                    ;IF EQ THEN CONTINUE
  6429
  6430 033764  013737  002326  002736          MOV     R.CS1,FP.ACT            ;LOAD ACTUAL VALUE
  6431 033772                                  ERRHRD  4,M.CSR2,ER.REG         ;''CS1 REGISTER STATUS ERROR''
       033772  104456                          TRAP    C$ERHRD
       033774  000004                          .WORD   4
       033776  007237                          .WORD   M.CSR2
       034000  015016                          .WORD   ER.REG
  6432
  6433 034002                          400$:   ENDSUB
       034002                          L10120:
       034002  104403                          TRAP    C$ESUB
```

```
 6435 034004                            STARS
                                        ;;*****************************************************************
 6436                                   ;;
 6437                                   ;;    SUBTEST 2:
 6438                                   ;;
 6439                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 6440                                   ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERROR
 6441                                   ;;      ISSUE INTERFACE READ CMD AND SET CYCLE REQUEST BIT
 6442                                   ;;      DELAY A FEW CYCLES AND SET IS REGISTER XABT BIT
 6443                                   ;;      READ REGISTERS, CHECK IRY BIT SET
 6444                                   ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
 6445 034004                            STARS
                                        ;;*****************************************************************
 6446
 6447 034004                                    BGNSUB
      034004                            T18.2:
      034004  104402                           TRAP    C$BSUB
 6448
 6449 034006  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6450 034012  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6451 034016  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6452
 6453 034020                                    ERRDF   4,M.MBC1,ER.DMP         ;DEVICE FATAL
      034020  104455                           TRAP    C$ERDF
      034022  000004                           .WORD   4
      034024  011650                           .WORD   M.MBC1
      034026  015074                           .WORD   ER.DMP
 6454 034030                                    DOCLN                           ;CLEAN UP AND EXIT
      034030  104444                           TRAP    C$DCLN
 6455
 6456 034032  012700  177740            100$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
 6457 034036  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 6458 034042  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 6459 034046  005737  003030                    TST     MBCFLG                  ;TEST FOR ERROR
 6460 034052  001405                            BEQ     150$                    ;BRANCH IF NO ERROR
 6461
 6462 034054                                    ERRHRD  5,M.BYT2,ER.DMP         ;''XFER ERROR ON INTERFACE WRITE''
      034054  104456                           TRAP    C$ERHRD
      034056  000005                           .WORD   5
      034060  007533                           .WORD   M.BYT2
      034062  015074                           .WORD   ER.DMP
 6463 034064  000463                           BR      400$                    ;EXIT SUBTEST
```

TEST 18: CHECK BYTE TRANSFER ABORT LOGIC (XABT)

```
6465
6466 034066  012700  177600          150$:  MOV     #-200,R0              ;LOAD BLOCK SIZE = 256 BYTES
6467 034072  004737  015726                 JSR     PC,RDINI              ;AND INITIALIZE ADDRESS REGISTERS
6468 034076  012777  000171  146172         MOV     #RIE.CMD,@A.CS1       ;ISSUE READ COMMAND TO CS1
6469 034104  004737  016030                 JSR     PC,SILODLY            ;WAIT FOR SILO TO FILL
6470 034110  012777  000020  146172         MOV     #IS.CYC,@A.IS         ;SET IS REGISTER CYC BIT
6471 034116  004737  016016                 JSR     PC,DMADLY             ;DELAY A FEW CYCLES
6472
6473 034122  012777  004000  146160         MOV     #IS.XABT,@A.IS        ;SET IS REGISTER TRANSFER ABORT
6474 034130  004737  015444                 JSR     PC,TSTINT             ;WAIT FOR INTERRUPT OR TIME-OUT
6475 034134  004737  015606                 JSR     PC,RDREG              ;READ REGISTERS
6476
6477 034140  032737  000200  002340         BIT     #IS.IRY,R.IS          ;TEST FOR IRY BIT SET
6478 034146  001005                         BNE     200$                  ;YES:  CONTINUE
6479
6480 034150                                 ERRHRD  6,M.IRY8,ER.DMP       ;''IRY NOT SET BY BYTE TRANSFER ABORT''
     034150  104456                         TRAP    C$ERHRD
     034152  000006                         .WORD   6
     034154  011500                         .WORD   M.IRY8
     034156  015074                         .WORD   ER.DMP
6481 034160  000425                         BR      400$                  ;EXIT SUBTEST
6482
6483 034162  005737  002330          200$:  TST     R.WC                  ;TEST WORD COUNT
6484 034166  100404                         BMI     300$                  ;IF LESS THAN ZERO THEN OK
6485
6486 034170                                 ERRHRD  7,M.ABT4,ER.DMP       ;''NO XFER ABORT ON INTERFACE READ''
     034170  104456                         TRAP    C$ERHRD
     034172  000007                         .WORD   7
     034174  006572                         .WORD   M.ABT4
     034176  015074                         .WORD   ER.DMP
6487
6488 034200  012737  144270  002734  300$:  MOV     #144270,FP.EXP        ;EXPECTED VALUE:  SC,TRE,DVA,RDY,READ
6489 034206  023737  002734  002326         CMP     FP.EXP,R.CS1          ;COMPARE CS1 WITH EXPECTED STATUS
6490 034214  001407                         BEQ     400$                  ;IF EQ THEN CONTINUE
6491
6492 034216  013737  002326  002736         MOV     R.CS1,FP.ACT          ;LOAD ACTUAL VALUE
6493 034224                                 ERRHRD  8,M.CSR2,ER.REG       ;''CS1 REGISTER STATUS ERROR''
     034224  104456                         TRAP    C$ERHRD
     034226  000010                         .WORD   8
     034230  007237                         .WORD   M.CSR2
     034232  015016                         .WORD   ER.REG
6494
6495 034234                         400$:   ENDSUB
     034234                         L10121:
     034234  104403                         TRAP    C$ESUB
6496
6497 034236  004737  015324                 JSR     PC,CLRISR             ;CLEAR VECTOR
6498
6499 034242  004737  015550                 JSR     PC,CLRTST             ;AND CLEAR TEST DATA
6500
6501 034246                                 ENDTST
     034246                         L10117:
     034246  104401                         TRAP    C$ETST
```

```
                                .SBTTL  TEST 19:  CHECK BYTE TRANSFER ABORT LOGIC (AT03)
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      CHECK BYTE DATA TRANSFER ABORT LOGIC THROUGH LOOP-BACK
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT REVA AND BYTE MODE AND AT3 ABORT NOT DISABLED THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          INITIALIZE 64-BYTE BLOCK SIZE AND
                                ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER CLEAR ERROR
                                ;;      NO WRITE/READ TRANSFER ABORT
                                ;;      TRE NOT SET:  ADJUST BANDWIDTH
                                ;;      INTERFACE IRY BIT NOT SET BY TRANSFER ABORT
                                STARS
                                ;;*****************************************************************

                                        BGNTST
034250                          T19::

                        000023  SELMASK =       B.UPAR + B.AT3 + B.AT0
                        000010  SELTEST =       B.BYTE

034250  013700  003010          MOV     SELECT,R0               ;READ SELECT WORD
034254  042700  000023          BIC     #SELMASK,R0             ;CLEAR DON'T CARE SELECT BITS
034260  022700  000010          CMP     #SELTEST,R0             ;TEST BYTE MODE SET
034264  001402                  BEQ     50$                     ;BRANCH IF EQUAL

034266                  25$:    EXIT    TST                     ;EXIT TEST
034266  104432                  TRAP    C$EXIT
034270  000476                  .WORD   L10122-.

034272  004737  015274  50$:    JSR     PC,SETISR               ;SETUP ISR VECTOR
034276  013737  002406  002730  MOV     NA.CS1,FP.NAM           ;LOAD REGISTER NAME ADDRESS
034304  013737  002436  002746  MOV     BA.CS1,FP.TBL           ;LOAD REGISTER BIT TABLE ADDRESS
                                                                ;EXECUTE SUBTEST
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 1:
                                        ;;
                                        ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      ISSUE INTERFACE WRITE CMD AND SET CYCLE REQUEST BIT
                                        ;;      DELAY A FEW CYCLES AND SET FS REGISTER FUNCTION BIT F5
                                        ;;      READ REGISTERS, CHECK IRY BIT SET
                                        ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB
                                        T19.1:
034312  104402                                  TRAP    C$BSUB

034314  004737  015342                50$:      JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
034320  005737  003030                          TST     MBCFLG                  ;TEST FLAG FOR ERROR
034324  001405                                  BEQ     100$                    ;CONTINUE IF NO ERROR

034326                                          ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
034326  104455                                  TRAP    C$ERDF
034330  000001                                  .WORD   1
034332  011650                                  .WORD   M.MBC1
034334  015074                                  .WORD   ER.DMP
034336                                          DOCLN                           ;CLEAN UP AND EXIT
034336  104444                                  TRAP    C$DCLN

034340  012700  177600          100$:     MOV     #-200,R0                ;LOAD BLOCK SIZE = 256 BYTES
034344  004737  015762                    JSR     PC,WRTINI               ;AND INITIALIZE ADDRESS REGISTERS
034350  012777  000161  145720            MOV     #WIE.CMD,@A.CS1         ;ISSUE WRITE COMMAND TO CS1
034356  004737  016030                    JSR     PC,SILODLY              ;WAIT FOR SILO TO FILL
034362  012777  000020  145720            MOV     #IS.CYC,@A.IS           ;SET CYCLE REQUEST
034370  004737  016016                    JSR     PC,DMADLY               ;DELAY A FEW CYCLES

034374  012777  004000  145702            MOV     #FS.F3,@A.FS            ;SET FS REGISTER FUNCTION BIT F3
034402  004737  015444                    JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME-OUT
034406  004737  015606                    JSR     PC,RDREG                ;READ REGISTERS
034412  012777  000000  145664            MOV     #0,@A.FS                ;CLEAR FUNCTION / STATUS BITS

034420  032737  000200  002340            BIT     #IS.IRY,R.IS            ;TEST FOR IRY BIT SET
034426  001005                            BNE     200$                    ;YES:  CONTINUE

034430                                  ERRHRD  2,M.IRY8,ER.DMP         ;"IRY NOT SET BY BYTE TRANSFER ABORT"
034430  104456                          TRAP    C$ERHRD
034432  000002                          .WORD   2
034434  011500                          .WORD   M.IRY8
034436  015074                          .WORD   ER.DMP
034440  000425                          BR      400$                    ;EXIT SUBTEST

034442  005737  002330          200$:     TST     R.WC                    ;TEST WORD COUNT
034446  100404                          BMI     300$                    ;IF LESS THAN ZERO THEN OK

034450                                  ERRHRD  3,M.ABT3,ER.DMP         ;"NO XFER ABORT ON INTERFACE WRITE"
034450  104456                          TRAP    C$ERHRD
034452  000003                          .WORD   3
```

```
        034454  006520                          .WORD   M.ABT3
        034456  015074                          .WORD   ER.DMP
   6584
   6585 034460  012737  144260  002734  300$:   MOV     #144260,FP.EXP          ;EXPECTED VALUE:  SC,TRE,DVA,RDY,WRITE
   6586 034466  023737  002734  002326          CMP     FP.EXP,R.CS1            ;COMPARE CS1 WITH EXPECTED STATUS
   6587 034474  001407                          BEQ     400$                    ;IF EQ THEN CONTINUE
   6588
   6589 034476  013737  002326  002736          MOV     R.CS1,FP.ACT            ;LOAD ACTUAL VALUE
   6590 034504                                  ERRHRD  4,M.CSR2,ER.REG         ;''CS1 REGISTER STATUS ERROR''
        034504  104456                          TRAP    C$ERHRD
        034506  000004                          .WORD   4
        034510  007237                          .WORD   M.CSR2
        034512  015016                          .WORD   ER.REG
   6591
   6592 034514                          400$:   ENDSUB
        034514                          L10123:
        034514  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERROR
                                        ;;      ISSUE INTERFACE READ CMD AND SET CYCLE REQUEST BIT
                                        ;;      DELAY A FEW CYCLES AND SET FS REGISTER FUNCTION BIT F5
                                        ;;      READ REGISTERS, CHECK IRY BIT SET
                                        ;;      CHECK WORD COUNT LESS THAN ZERO, AND TRE BIT SET
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB
                                        T19.2:
                                                TRAP    C$BSUB

034520  004737  015342        50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
034524  005737  003030                TST     MBCFLG                  ;TEST FLAG FOR ERROR
034530  001405                        BEQ     100$                    ;CONTINUE IF NO ERROR

034532                                  ERRDF   4,M.MBC1,ER.DMP         ;DEVICE FATAL
034532  104455                          TRAP    C$ERDF
034534  000004                          .WORD   4
034536  011650                          .WORD   M.MBC1
034540  015074                          .WORD   ER.DMP
034542                                  DOCLN                           ;CLEAN UP AND EXIT
034542  104444                          TRAP    C$DCLN

034544  012700  177740        100$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 64 BYTES
034550  004737  015674                JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
034554  004737  015410                JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
034560  005737  003030                TST     MBCFLG                  ;TEST FOR ERROR
034564  001405                        BEQ     150$                    ;BRANCH IF NO ERROR

034566                                  ERRHRD  5,M.BYT2,ER.DMP         ;"XFER ERROR ON INTERFACE WRITE"
034566  104456                          TRAP    C$ERHRD
034570  000005                          .WORD   5
034572  007533                          .WORD   M.BYT2
034574  015074                          .WORD   ER.DMP
034576  000466                        BR      400$                    ;EXIT SUBTEST
```

```
 6624
 6625 034600  012700  177600          150$:  MOV     #-200,R0              ;LOAD BLOCK SIZE = 256 BYTES
 6626 034604  004737  015726                 JSR     PC,RDINI              ;AND INITIALIZE ADDRESS REGISTERS
 6627 034610  012777  000171  145460         MOV     #RIE.CMD,@A.CS1       ;ISSUE READ COMMAND TO CS1
 6628 034616  004737  016030                 JSR     PC,SILODLY            ;WAIT FOR SILO TO FILL
 6629 034622  012777  000020  145460         MOV     #IS.CYC,@A.IS         ;SET CYCLE REQUEST
 6630 034630  004737  016016                 JSR     PC,DMADLY             ;DELAY A FEW CYCLES
 6631
 6632 034634  012777  004000  145442         MOV     #FS.F3,@A.FS          ;SET FS REGISTER FUNCTION BIT F3
 6633 034642  004737  015444                 JSR     PC,TSTINT             ;WAIT FOR INTERRUPT OR TIME-OUT
 6634 034646  004737  015606                 JSR     PC,RDREG              ;READ REGISTERS
 6635 034652  012777  000000  145424         MOV     #0,@A.FS              ;CLEAR FUNCTION / STATUS BITS
 6636
 6637 034660  032737  000200  002340         BIT     #IS.IRY,R.IS          ;TEST FOR IRY BIT SET
 6638 034666  001005                         BNE     200$                  ;YES:  CONTINUE
 6639
 6640 034670                                 ERRHRD  6,M.IRY8,ER.DMP       ;''IRY NOT SET BY BYTE TRANSFER ABORT''
      034670  104456                         TRAP    C$ERHRD
      034672  000006                         .WORD   6
      034674  011500                         .WORD   M.IRY8
      034676  015074                         .WORD   ER.DMP
 6641 034700  000425                         BR      400$                  ;EXIT SUBTEST
 6642
 6643 034702  005737  002330          200$:  TST     R.WC                  ;TEST WORD COUNT
 6644 034706  100404                         BMI     300$                  ;IF LESS THAN ZERO THEN OK
 6645
 6646 034710                                 ERRHRD  7,M.ABT4,ER.DMP       ;''NO XFER ABORT ON INTERFACE READ''
      034710  104456                         TRAP    C$ERHRD
      034712  000007                         .WORD   7
      034714  006572                         .WORD   M.ABT4
      034716  015074                         .WORD   ER.DMP
 6647
 6648 034720  012737  144270  002734  300$:  MOV     #144270,FP.EXP        ;EXPECTED VALUE:  SC.TRE,DVA,RDY,READ
 6649 034726  023737  002734  002326         CMP     FP.EXP,R.CS1          ;COMPARE CS1 WITH EXPECTED STATUS
 6650 034734  001407                         BEQ     400$                  ;IF EQ THEN CONTINUE
 6651
 6652 034736  013737  002326  002736         MOV     R.CS1,FP.ACT          ;LOAD ACTUAL VALUE
 6653 034744                                 ERRHRD  8,M.CSR2,ER.REG       ;''CS1 REGISTER STATUS ERROR''
      034744  104456                         TRAP    C$ERHRD
      034746  000010                         .WORD   8
      034750  007237                         .WORD   M.CSR2
      034752  015016                         .WORD   ER.REG
 6654
 6655 034754                         400$:   ENDSUB
      034754                         L10124:
      034754  104403                         TRAP    C$ESUB
 6656
 6657 034756  004737  015324                 JSR     PC,CLRISR             ;CLEAR ISR VECTOR
 6658
 6659 034762  004737  015550                 JSR     PC,CLRTST             ;AND CLEAR TEST DATA
 6660
 6661 034766                                 ENDTST
      034766                         L10122:
      034766  104401                         TRAP    C$ETST
```

```
                                .SBTTL  TEST 20:  CHECK INTERFACE STATUS ICP BIT
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      VERIFY PROPER FUNCTION OF IS REGISTER ICP BIT
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT REVA THEN:
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                ;;          SETUP ISR VECTOR AND DISPLAY PARMS
                                ;;          EXECUTE SUBTEST 1-2 AND CLEAR VECTOR
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      IS REGISTER ICP BIT NOT SET/RESET
                                ;;
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T20::
                    000200      SELTEST =       B.REVA                  ;TEST SELECT BIT

034770  013700  003010                  MOV     SELECT,R0               ;LOAD TEST SELECT WORD
034774  032700  000200                  BIT     #SELTEST,R0             ;TEST FOR REVA SET
035000  001402                        BEQ     50$                     ;CONTINUE IF REVA NOT SET

035002                                  EXIT    TST                     ;EXIT FROM TEST
035002  104432                        TRAP    C$EXIT
035004  000330                        .WORD   L10125-.

035006  004737  015274        50$:    JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
035012  013737  002420  002730        MOV     NA.IS,FP.NAM            ;LOAD IS REGISTER NAME POINTER
035020  013737  002450  002746        MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                        ;EXECUTE SUBTEST
```

```
 6697 035026                            STARS
                                        ;;*****************************************************************
 6698                                   ;;
 6699                                   ;;    SUBTEST 1:
 6700                                   ;;
 6701                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 6702                                   ;;      SET PAT BIT IN CS2 REGISTER AND ISSUE INTERFACE WRITE CMD
 6703                                   ;;      READ REGISTERS AND TEST INTERFACE STATUS ICP SET
 6704                                   ;;      RESET CS2 REGISTER PAT BIT AND ISSUE DRIVE CLEAR COMMAND
 6705                                   ;;      READ REGISTERS AND TEST INTERFACE STATUS ICP RESET
 6706 035026                            STARS
                                        ;;*****************************************************************
 6707
 6708 035026                                    BGNSUB
      035026                            T20.1:
      035026  104402                            TRAP    C$BSUB
 6709
 6710 035030  004737  015342            50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6711 035034  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6712 035040  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6713
 6714 035042                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      035042  104455                            TRAP    C$ERDF
      035044  000001                            .WORD   1
      035046  011650                            .WORD   M.MBC1
      035050  015074                            .WORD   ER.DMP
 6715 035052                                    DOCLN                           ;CLEAN UP AND EXIT
      035052  104444                            TRAP    C$DCLN
 6716
 6717 035054  052777  000020  145224  100$:    BIS     #CS2.PAT,@A.CS2         ;SET PARITY TEST
 6718 035062  012700  177777                    MOV     #-1,R0                  ;LOAD BLOCK SIZE = 1 WORD
 6719 035066  004737  015674                    JSR     PC,WRTBLK               ;ISSUE INTERFACE WRITE CMD
 6720 035072  004737  015606                    JSR     PC,RDREG                ;AND READ REGISTERS
 6721
 6722 035076  032737  001000  002340            BIT     #IS.ICP,R.IS            ;TEST ICP BIT SET
 6723 035104  001004                            BNE     200$                    ;BRANCH IF SET
 6724
 6725 035106                                    ERRHRD  2,M.ICP1,ER.DMP         ;"ICP BIT NOT SET BY PARITY TEST"
      035106  104456                            TRAP    C$ERHRD
      035110  000002                            .WORD   2
      035112  010450                            .WORD   M.ICP1
      035114  015074                            .WORD   ER.DMP
 6726
 6727 035116  042777  000020  145162  200$:    BIC     #CS2.PAT,@A.CS2         ;CLEAR PARITY TEST
 6728 035124  004737  015464                    JSR     PC,CLRDRI               ;INTERFACE CLEAR CMD & READ REGISTERS
 6729 035130  005737  003032                    TST     DRIFLG                  ;TEST FOR INTERFACE ERROR
 6730 035134  001405                            BEQ     300$                    ;CONTINUE IF NO ERROR
 6731
 6732 035136                                    ERRHRD  3,M.ISR2,ER.DMP         ;"INTERFACE STATUS ERROR ON CLEAR"
      035136  104456                            TRAP    C$ERHRD
      035140  000003                            .WORD   3
      035142  011572                            .WORD   M.ISR2
      035144  015074                            .WORD   ER.DMP
 6733 035146  000410                            BR      400$                    ;EXIT TEST
 6734
 6735 035150  032737  001000  002340  300$:    BIT     #IS.ICP,R.IS            ;TEST ICP BIT SET
 6736 035156  001404                            BEQ     400$                    ;BRANCH IF RESET
```

```
 6737
 6738 035160                                ERRHRD  4,M.ICP2,ER.DMP         ;''ICP BIT NOT RESET BY DRIVE CLEAR''
      035160  104456                        TRAP    C$ERHRD
      035162  000004                        .WOPD   4
      035164  010535                        .WORD   M.ICP2
      035166  015074                        .WORD   ER.DMP
 6739
 6740 035170                          400$:  ENDSUB                          ;SET ICP
      035170                          L10126:
      035170  104403                        TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      SET PAT BIT IN CS2 REGISTER AND ISSUE INTERFACE WRITE CMD
                                        ;;      READ REGISTERS AND TEST INTERFACE STATUS ICP SET
                                        ;;      RESET CS2 REGISTER PAT BIT AND WRITE TO INTERFACE STATUS ICP
                                        ;;      READ REGISTERS AND TEST INTERFACE STATUS ICP RESET
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB
                                        T20.2:
                                                TRAP    C$BSUB
                                        50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
                                                TST     MBCFLG                  ;TEST FLAG FOR ERROR
                                                BEQ     100$                    ;CONTINUE IF NO ERROR

                                                ERRDF   5,M.MBC1,ER.DMP         ;DEVICE FATAL
                                                TRAP    C$ERDF
                                                .WORD   5
                                                .WORD   M.MBC1
                                                .WORD   ER.DMP
                                                DOCLN                           ;CLEAN UP AND EXIT
                                                TRAP    C$DCLN

                                        100$:   BIS     #CS2.PAT,@A.CS2         ;SET PARITY TEST
                                                MOV     #-1,R0                  ;LOAD BLOCK SIZE = 1 WORD
                                                JSR     PC,WRTBLK               ;ISSUE INTERFACE WRITE CMD
                                                JSR     PC,RDREG                ;AND READ REGISTERS

                                                BIT     #IS.ICP,R.IS            ;TEST ICP BIT SET
                                                BNE     200$                    ;BRANCH IF SET

                                                ERRHRD  6,M.ICP1,ER.DMP         ;"ICP BIT NOT SET BY PARITY TEST"
                                                TRAP    C$ERHRD
                                                .WORD   6
                                                .WORD   M.ICP1
                                                .WORD   ER.DMP

                                        200$:   BIC     #CS2.PAT,@A.CS2         ;CLEAR PARITY TEST
                                                BIS     #IS.ICP,@A.IS           ;WRITE TO INTERFACE STATUS ICP
                                                JSR     PC,RDREG                ;READ REGISTERS
                                                BIT     #IS.ICP,R.IS            ;TEST FOR ICP RESET
                                                BEQ     400$                    ;CONTINUE IF NO ERROR

                                                ERRHRD  7,M.ICP3,ER.DMP         ;"INTERFACE STATUS ICP NOT RESET"
                                                TRAP    C$ERHRD
                                                .WORD   7
                                                .WORD   M.ICP3
                                                .WORD   ER.DMP

                                        400$:   ENDSUB                          ;RESET ICP
                                        L10127:
```

TEST 20: CHECK INTERFACE STATUS ICP BIT

```
        035322  104403                          TRAP    C$ESUB
  6781
  6782 035324  004737  015324                   JSR     PC,CLRISR           ;CLEAR SERVICE ROUTINE VECTOR
  6783
  6784 035330  004737  015550                   JSR     PC,CLRTST           ;AND CLEAR TEST DATA
  6785
  6786 035334                                   ENDTST
       035334                          L10125:
       035334  104401                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 21:  CHECK INTERFACE STATUS IDP BIT
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;  TEST DESCRIPTION:
                                        ;;
                                        ;;      VERIFY PROPER FUNCTION OF IS REGISTER IDP BIT
                                        ;;
                                        ;;    TEST STEPS:
                                        ;;
                                        ;;      IF NOT REVA OR NOT UPAR THEN:
                                        ;;          SETUP ISR VECTOR AND INITIALIZE
                                        ;;          EXECUTE SUBTEST 1-4 AND CLEAR VECTOR
                                        ;;
                                        ;;    ERROR CONDITIONS:
                                        ;;
                                        ;;      IS REGISTER IDP BIT NOT SET/RESET
                                        STARS
                                        ;;*****************************************************************

                                            BGNTST
035336                                  T21::
                 000017                 SELMASK =       B.BYTE + B.ABORT + B.AT3 + B.AT0

035336  013700  003010                        MOV     SELECT,R0               ;LOAD TEST SELECT WORD
035342  042700  000017                        BIC     #SELMASK,R0             ;TEST FOR REVA OR UPAR SET
035346  001402                              BEQ     50$                     ;CONTINUE IF REVA AND UPAR NOT SET

035350                                      EXIT    TST                     ;EXIT FROM TEST
035350  104432                              TRAP    C$EXIT
035352  001232                              .WORD   L10130-.

035354  004737  015274          50$:        JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
035360  013737  002420  002730              MOV     NA.IS,FP.NAM            ;LOAD IS REGISTER NAME POINTER
035366  013737  002450  002746              MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
                                                                            ;EXECUTE SUBTEST
```

```
 6821 035374                            STARS
                                        ;;*****************************************************************
 6822                                   ;;
 6823                                   ;;    SUBTEST 1:
 6824                                   ;;
 6825                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 6826                                   ;;      SETUP DATA PATTERN WITH ODD LOW BYTE, EVEN HIGH BYTE
 6827                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 6828                                   ;;      SET CYCLE REQUEST, INVERT PARITY AND WAIT FOR INTERRUPT
 6829                                   ;;      READ REGISTERS AND CHECK IDP & ERR BITS SET
 6830                                   ;;      WRITE INTERFACE CLEAR CMD, AND CHECK IDP BIT RESET
 6831 035374                            STARS
                                        ;;*****************************************************************
 6832
 6833 035374                                    BGNSUB
      035374                            T21.1:
      035374  104402                           TRAP    C$BSUB
 6834
 6835 035376  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6836 035402  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6837 035406  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6838
 6839 035410                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      035410  104455                           TRAP    C$ERDF
      035412  000001                           .WORD   1
      035414  011650                           .WORD   M.MBC1
      035416  015074                           .WORD   ER.DMP
 6840 035420                                    DOCLN                           ;CLEAN UP AND EXIT
      035420  104444                           TRAP    C$DCLN
 6841
 6842 035422  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 6843 035424  012701  000001                    MOV     #1,R1                   ;LOAD DATA PATTERN
 6844 035430  010137  002744                    MOV     R1,FP.DPAT              ;AND INITIALIZE DISP PARM
 6845
 6846 035434  010162  004054            150$:   MOV     R1,OBUF(R2)             ;LOAD EVEN HIGH BYTE ODD LOW BYTE
 6847 035440  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 6848 035446  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 6849 035452  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6850 035456  003366                            BCT     150$                    ;CONTINUE UNTIL BUFFER INITIALIZED
 6851
 6852 035460  012700  177740            200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
 6853 035464  004737  015762                    JSR     PC,WRTINI               ;INITIALIZE FOR INTERFACE WRITE
 6854 035470  012777  000161  144600            MOV     #WIE.CMD,@A.CS1         ;ISSUE INTERFACE WRITE
 6855 035476  012777  010020  144604            MOV     #10020,@A.IS            ;CYCLE REQUEST WITH INVERT PARITY
 6856 035504  004737  015444                    JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME OUT
 6857
 6858 035510  004737  015606                    JSR     PC,RDREG                ;READ REGISTERS
 6859 035514  032737  000400  002340            BIT     #IS.IDP,R.IS            ;TEST IDP BIT SET
 6860 035522  001014                            BNE     300$                    ;BRANCH IF SET
 6861
 6862 035524  013737  002340  002736            MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
 6863 035532  012737  150600  002734            MOV     #150600,FP.EXP          ;EXPECTED:  ATA, ERR, IP, IDP & IRY
 6864 035540                                    ERRHRD  2,M.IDP1,ER.REG         ;"IDP BIT NOT SET BY PARITY ERROR"
      035540  104456                           TRAP    C$ERHRD
      035542  000002                           .WORD   2
      035544  010215                           .WORD   M.IDP1
      035546  015016                           .WORD   ER.REG
```

```
   6865
   6866 035550  004737  016510                JSR     PC,DSPDPAT              ;DISPLAY DATA PATTERN
```

TEST 21: CHECK INTERFACE STATUS IDP BIT

```
6868
6869 035554  004737  015464          300$:   JSR     PC,CLRDRI               ;ISSUE INTERFACE CLEAR
6870 035560  005737  003032                  TST     DRIFLG                  ;TEST ERROR FLAG
6871 035564  001405                          BEQ     400$                    ;BRANCH IF OK
6872
6873 035566                                   ERRHRD  3,M.ISR2,ER.DMP         ;"INTERFACE STATUS ERROR ON CLEAR"
     035566  104456                          TRAP    C$ERHRD
     035570  000003                          .WORD   3
     035572  011572                          .WORD   M.ISR2
     035574  015074                          .WORD   ER.DMP
6874 035576  000416                          BR      500$                    ;EXIT TEST
6875
6876 035600  032737  000400  002340  400$:   BIT     #IS.IDP,R.IS            ;TEST FOR IDP BIT RESET
6877 035606  001412                          BEQ     500$                    ;BRANCH IF RESET
6878
6879 035610  013737  002340  002736          MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
6880 035616  012737  000200  002734          MOV     #200,FP.EXP             ;EXPECTED:  IRY
6881 035624                                   ERRHRD  4,M.IDP2,ER.REG         ;"IDP NOT RESET BY DRIVE CLEAR"
     035624  104456                          TRAP    C$ERHRD
     035626  000004                          .WORD   4
     035630  010277                          .WORD   M.IDP2
     035632  015016                          .WORD   ER.REG
6882
6883 035634                          500$:   ENDSUB
     035634                          L10131:
     035634  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 2:
                                        ;;
                                        ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERROR
                                        ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
                                        ;;      SET CYCLE REQUEST, INVERT PARITY AND WAIT FOR INTERRUPT
                                        ;;      READ REGISTERS AND CHECK IDP & ERR BITS SET
                                        ;;      WRITE TO INTERFACE STATUS IDP, READ REGISTERS AND CHECK IDP RESET
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB
                                        T21.2:
035636  104402                                  TRAP    C$BSUB

035640  004737  015342                          JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
035644  005737  003030                          TST     MBCFLG                  ;TEST FLAG FOR ERROR
035650  001405                                  BEQ     200$                    ;CONTINUE IF NO ERROR

035652                                          ERRDF   5,M.MBC1,ER.DMP         ;DEVICE FATAL
035652  104455                                  TRAP    C$ERDF
035654  000005                                  .WORD   5
035656  011650                                  .WORD   M.MBC1
035660  015074                                  .WORD   ER.DMP
035662                                          DOCLN                           ;CLEAN UP AND EXIT
035662  104444                                  TRAP    C$DCLN

035664  012700  177740          200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
035670  004737  015674                  JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
035674  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
035700  005737  003030                  TST     MBCFLG                  ;TEST ERROR FLAG
035704  001404                          BEQ     225$                    ;BRANCH IF NO ERROR

035706                                  ERRHRD  6,M.WRD2,ER.DMP         ;"XFER ERROR ON INTERFACE WRITE"
035706  104456                          TRAP    C$ERHRD
035710  000006                          .WORD   6
035712  007324                          .WORD   M.WRD2
035714  015074                          .WORD   ER.DMP

035716  012700  177740          225$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
035722  004737  015726                  JSR     PC,RDINI                ;INITIALIZE FOR INTERFACE READ
035726  012777  000171  144342          MOV     #RIE.CMD,@A.CS1         ;ISSUE INTERFACE READ
035734  012777  010020  144346          MOV     #10020,@A.IS            ;CYCLE REQUEST WITH INVERT PARITY
035742  004737  015444                  JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME OUT

035746  004737  015606          250$:   JSR     PC,RDREG                ;READ REGISTERS
035752  032737  000400  002340          BIT     #IS.IDP,R.IS            ;TEST IDP BIT SET
035760  001026                          BNE     400$                    ;BRANCH IF SET

035762  013737  002340  002736          MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
035770  012737  150600  002734          MOV     #150600,FP.EXP          ;EXPECTED:  ATA, ERR, IP, IDP & IRY
035776                                  ERRHRD  7,M.IDP1,ER.REG         ;"IDP BIT NOT SET BY PARITY ERROR"
035776  104456                          TRAP    C$ERHRD
036000  000007                          .WORD   7
```

TEST 21: CHECK INTERFACE STATUS IDP BIT

```
        036002  010215                          .WORD   M.IDP1
        036004  015016                          .WORD   ER.REG
   6927
   6928 036006  004737  016510                  JSR     PC,DSPDPAT              ;DISPLAY DATA PATTERN
   6929
   6930 036012                                  PRINTX  #F.PERR                 ;PROMPT FOR PARITY SWITCH
        036012  012746  013412                  MOV     #F.PERR,-(SP)
        036016  012746  000001                  MOV     #1,-(SP)
        036022  010600                          MOV     SP,R0
        036024  104415                          TRAP    C$PNTX
        036026  062706  000004                  ADD     #4,SP
   6931 036032  005237  003052                  INC     PERFLG                  ;BUMP MESSAGE COUNT
```

TEST 21: CHECK INTERFACE STATUS IDP BIT

```
6933
6934 036036  052777  C00400  144244  400$:   BIS     #IS.IDP,@A.IS           ;WRITE TO INTERFACE STATUS IDP
6935 036044  004737  015606                  JSR     PC,RDREG                ;READ REGISTERS AND
6936 036050  032737  000400  002340          BIT     #IS.IDP,R.IS            ;TEST FOR IDP BIT RESET
6937 036056  001412                          BEQ     500$                    ;BRANCH IF RESET
6938
6939 036060  013737  002340  002736          MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
6940 036066  012737  000200  002734          MOV     #200,FP.EXP             ;EXPECTED:  IRY
6941 036074                                  ERRHRD  8,M.IDP3,ER.REG         ;"IDP NOT RESET BY DRIVE CLEAR"
     036074  104456                          TRAP    C$ERHRD
     036076  000010                          .WORD   8
     036100  010365                          .WORD   M.IDP3
     036102  015016                          .WORD   ER.REG
6942
6943 036104                          500$:   ENDSUB
     036104                          L10132:
     036104  104403                          TRAP    C$ESUB
```

```
 6945 036106                            STARS
                                        ;;*****************************************************************
 6946                                   ;;
 6947                                   ;;    SUBTEST 3:
 6948                                   ;;
 6949                                   ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
 6950                                   ;;      SETUP DATA PATTERN WITH ODD LOW BYTE, EVEN HIGH BYTE
 6951                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 6952                                   ;;      SET CYCLE REQUEST, INVERT PARITY AND WAIT FOR INTERRUPT
 6953                                   ;;      READ REGISTERS AND CHECK IDP & ERR BITS SET
 6954                                   ;;      WRITE INTERFACE CLEAR CMD, AND CHECK IDP BIT RESET
 6955 036106                            STARS
                                        ;;*****************************************************************
 6956
 6957 036106                                    BGNSUB
      036106                            T21.3:
      036106  104402                           TRAP    C$BSUB
 6958
 6959 036110  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 6960 036114  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 6961 036120  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 6962
 6963 036122                                    ERRDF   9,M.MBC1,ER.DMP         ;DEVICE FATAL
      036122  104455                           TRAP    C$ERDF
      036124  000011                           .WORD   9
      036126  011650                           .WORD   M.MBC1
      036130  015074                           .WORD   ER.DMP
 6964 036132                                    DOCLN                           ;CLEAN UP AND EXIT
      036132  104444                           TRAP    C$DCLN
 6965
 6966 036134  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 6967 036136  012701  000400                    MOV     #400,R1                 ;LOAD DATA PATTERN
 6968 036142  010137  002744                    MOV     R1,FP.DPAT              ;AND INITIALIZE DISP PARM
 6969
 6970 036146  010162  004054            150$:   MOV     R1,OBUF(R2)             ;LOAD ODD HIGH BYTE EVEN LOW BYTE
 6971 036152  012762  000000  003054            MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
 6972 036160  062702  000002                    ADD     #2,R2                   ;AND BUMP INDEX
 6973 036164  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
 6974 036170  003366                            BGT     150$                    ;CONTINUE UNTIL BUFFER INITIALIZED
 6975
 6976 036172  012700  177740            200$:   MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
 6977 036176  004737  015762                    JSR     PC,WRTINI               ;INITIALIZE FOR INTERFACE WRITE
 6978 036202  012777  000161  144066            MOV     #WIE.CMD,@A.CS1         ;ISSUE INTERFACE WRITE
 6979 036210  012777  010020  144072            MOV     #10020,@A.IS            ;CYCLE REQUEST WITH INVERT PARITY
 6980 036216  004737  015444                    JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME OUT
 6981
 6982 036222  004737  015606                    JSR     PC,RDREG                ;READ REGISTERS
 6983 036226  032737  000400  002340            BIT     #IS.IDP,R.IS            ;TEST IDP BIT SET
 6984 036234  001014                            BNE     300$                    ;BRANCH IF SET
 6985
 6986 036236  013737  002340  002736            MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
 6987 036244  012737  150600  002734            MOV     #150600,FP.EXP          ;EXPECTED:  ATA, ERR, IP, IDP & IRY
 6988 036252                                    ERRHRD  10,M.IDP1,ER.REG        ;"IDP BIT NOT SET BY PARITY ERROR"
      036252  104456                           TRAP    C$ERHRD
      036254  000012                           .WORD   10
      036256  010215                           .WORD   M.IDP1
      036260  015016                           .WORD   ER.REG
```

```
 6989
 6990 036262  004737  016510                JSR     PC,DSPDPAT              ;DISPLAY DATA PATTERN
```

TEST 21: CHECK INTERFACE STATUS IDP BIT

```
 6992
 6993 036266  004737  015464          300$:   JSR     PC,CLRDRI               ;ISSUE INTERFACE CLEAR
 6994 036272  005737  003032                  TST     DRIFLG                  ;TEST ERROR FLAG
 6995 036276  001405                          BEQ     400$                    ;BRANCH IF OK
 6996
 6997 036300                                  ERRHRD  11,M.ISR2,ER.DMP        ;''INTERFACE STATUS ERROR ON CLEAR''
      036300  104456                          TRAP    C$ERHRD
      036302  000013                          .WORD   11
      036304  011572                          .WORD   M.ISR2
      036306  015074                          .WORD   ER.DMP
 6998 036310  000416                          BR      500$                    ;EXIT TEST
 6999
 7000 036312  032737  000400  002340  400$:   BIT     #IS.IDP,R.IS            ;TEST FOR IDP BIT RESET
 7001 036320  001412                          BEQ     500$                    ;BRANCH IF RESET
 7002
 7003 036322  013737  002340  002736          MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
 7004 036330  012737  000200  002734          MOV     #200,FP.EXP             ;EXPECTED:  IRY
 7005 036336                                  ERRHRD  12,M.IDP2,ER.REG        ;''IDP NOT RESET BY DRIVE CLEAR''
      036336  104456                          TRAP    C$ERHRD
      036340  000014                          .WORD   12
      036342  010277                          .WORD   M.IDP2
      036344  015016                          .WORD   ER.REG
 7006
 7007 036346                          500$:   ENDSUB
      036346                          L10133:
      036346  104403                          TRAP    C$ESUB
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;    SUBTEST 4:
                                        ;;
                                        ;;      CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                        ;;      WRITE BLOCK TO INTERFACE AND CHECK FOR ERROR
                                        ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
                                        ;;      SET CYCLE REQUEST, INVERT PARITY AND WAIT FOR INTERRUPT
                                        ;;      READ REGISTERS AND CHECK IDP & ERR BITS SET
                                        ;;      WRITE TO INTERFACE STATUS IDP, READ REGISTERS AND CHECK IDP BIT RESET
                                        ;;
                                        STARS
                                        ;;*****************************************************************

                                                BGNSUB
                                        T21.4:
036350  104402                                  TRAP    C$BSUB

036352  004737  015342                          JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
036356  005737  003030                          TST     MBCFLG                  ;TEST FLAG FOR ERROR
036362  001405                                  BEQ     200$                    ;CONTINUE IF NO ERROR

036364                                          ERRDF   13,M.MBC1,ER.DMP        ;DEVICE FATAL
036364  104455                                  TRAP    C$ERDF
036366  000015                                  .WORD   13
036370  011650                                  .WORD   M.MBC1
036372  015074                                  .WORD   ER.DMP
036374                                          DOCLN                           ;CLEAN UP AND EXIT
036374  104444                                  TRAP    C$DCLN

036376  012700  177740          200$:           MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
036402  004737  015674                          JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
036406  004737  015410                          JSR     PC,TSTMBC               ;READ REGISTERS AND TEST FOR ERROR
036412  005737  003030                          TST     MBCFLG                  ;TEST ERROR FLAG
036416  001404                                  BEQ     225$                    ;BRANCH IF NO ERROR

036420                                          ERRHRD  14,M.WRD2,ER.DMP        ;"XFER ERROR ON INTERFACE WRITE"
036420  104456                                  TRAP    C$ERHRD
036422  000016                                  .WORD   14
036424  007324                                  .WORD   M.WRD2
036426  015074                                  .WORD   ER.DMP

036430  012700  177740          225$:           MOV     #-40,R0                 ;LOAD BLOCK SIZE = 32 WORDS/BYTES
036434  004737  015726                          JSR     PC,RDINI                ;INITIALIZE FOR INTERFACE READ
036440  012777  000171  143630                  MOV     #RIE.CMD,@A.CS1         ;ISSUE INTERFACE READ
036446  012777  010020  143634                  MOV     #10020,@A.IS            ;CYCLE REQUEST WITH INVERT PARITY
036454  004737  015444                          JSR     PC,TSTINT               ;WAIT FOR INTERRUPT OR TIME OUT

036460  004737  015606                          JSR     PC,RDREG                ;READ REGISTERS
036464  032737  000400  002340                  BIT     #IS.IDP,R.IS            ;TEST IDP BIT SET
036472  001014                                  BNE     400$                    ;BRANCH IF SET

036474  013737  002340  002736                  MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
036502  012737  150600  002734                  MOV     #150600,FP.EXP          ;EXPECTED:  ATA, ERR, IP, IDP & IRY
036510                                          ERRHRD  15,M.IDP1,ER.REG        ;"IDP BIT NOT SET BY PARITY ERROR"
036510  104456                                  TRAP    C$ERHRD
036512  000017                                  .WORD   15
```

TEST 21: CHECK INTERFACE STATUS IDP BIT

```
        036514  010215                          .WORD   M.IDP1
        036516  015016                          .WORD   ER.REG
   7051
   7052 036520  004737  016510                  JSR     PC,DSPDPAT              ;DISPLAY DATA PATTERN
   7053
   7054 036524  052777  000400  143556  400$:   BIS     #IS.IDP,@A.IS           ;WRITE TO INTERFACE STATUS IDP
   7055 036532  004737  015606                  JSR     PC,RDREG                ;READ REGISTERS AND
   7056 036536  032737  000400  002340          BIT     #IS.IDP,R.IS            ;TEST FOR IDP BIT RESET
   7057 036544  001412                          BEQ     500$                    ;BRANCH IF RESET
   7058
   7059 036546  013737  002340  002736          MOV     R.IS,FP.ACT             ;LOAD ACTUAL VALUE
   7060 036554  012737  000200  002734          MOV     #200,FP.EXP             ;EXPECTED:  IRY
   7061 036562                                  ERRHRD  16,M.IDP3,ER.REG        ;"INTERFACE STATUS IDP NOT RESET"
        036562  104456                          TRAP    C$ERHRD
        036564  000020                          .WORD   16
        036566  010365                          .WORD   M.IDP3
        036570  015016                          .WORD   ER.REG
   7062
   7063 036572                          500$:   ENDSUB
        036572                          L10134:
        036572  104403                          TRAP    C$ESUB
   7064
   7065 036574  004737  015324                  JSR     PC,CLRISR               ;CLEAR SERVICE ROUTINE VECTOR
   7066
   7067 036600  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
   7068
   7069 036604                                  ENDTST
        036604                          L10130:
        036604  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 22:  VERIFY SW4 DISABLES INTERFACE AT00 BIT ATTN
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      VERIFY SW4 IN "OFF" POSITION WILL DISABLE
                                ;;      USER ATTENTION BIT AT00 FROM ASSERTING ATA
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT REVA AND AT0 DISABLED THEN:
                                ;;          CLEAR MBC AND CHECK FOR ERROR
                                ;;          ISSUE INTERFACE CLEAR CMD AND TEST FOR ERROR STATUS
                                ;;          SET FS REGISTER FUNCTION BIT F0
                                ;;          READ IS REGISTER AND CHECK AT00 BIT SET & ATA BIT RESET
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      IS REGISTER AT00 & ATA BITS NOT SET/RESET PROPERLY
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T22::

                    000036      SELMASK =       B.UPAR + B.BYTE + B.ABORT + B.AT3
                    000001      SELTEST =       B.AT0

036606  013700  003010                MOV     SELECT,R0               ;READ SELECT WORD
036612  042700  000036                BIC     #SELMASK,R0             ;CLEAR DON'T CARE SELECT BITS
036616  022700  000001                CMP     #SELTEST,R0             ;COMPARE AND
036622  001402                        BEQ     50$                     ;BRANCH IF EQUAL

036624                                EXIT    TST                     ;EXIT TEST
036624  104432                        TRAP    C$EXIT
036626  000164                        .WORD   L10135-.

036630  004737  015342        50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
036634  005737  003030                TST     MBCFLG                  ;TEST FLAG FOR ERROR
036640  001405                        BEQ     100$                    ;CONTINUE IF NO ERROR

036642                                ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
036642  104455                        TRAP    C$ERDF
036644  000001                        .WORD   1
036646  011650                        .WORD   M.MBC1
036650  015074                        .WORD   ER.DMP
036652                                DOCLN                           ;CLEAN UP AND EXIT
036652  104444                        TRAP    C$DCLN
```

```

 036654  004737  015464          100$:   JSR     PC,CLRDRI               ;CLEAR DRIVE AND TEST FOR ERROR
 036660  005737  003032                  TST     DRIFLG                  ;TEST ERROR FLAG
 036664  001405                          BEQ     150$                    ;CONTINUE IF NO ERROR

 036666                                  ERRHRD  2,M.ISR2,ER.DMP         ;"INTERFACE STATUS ERROR ON CLEAR CMD"
 036666  104456                          TRAP    C$ERHRD
 036670  000002                          .WORD   2
 036672  011572                          .WORD   M.ISR2
 036674  015074                          .WORD   ER.DMP
 036676  000443                          BR      400$                    ;EXIT TEST

 036700  012777  000400  143376  150$:   MOV     #FS.F0,@A.FS            ;SET FUNCTION BIT F5
 036706  017737  143376  002736          MOV     @A.IS,FP.ACT            ;READ REGISTER
 036714  012777  000000  143362          MOV     #0,@A.FS                ;CLEAR FS REGISTER

 036722  012737  000201  002734  200$:   MOV     #201,FP.EXP             ;LOAD EXPECTED:  IRY & AT00
 036730  032737  100000  002736          BIT     #IS.ATA,FP.ACT          ;TEST FOR ATA RESET
 036736  001413                          BEQ     300$                    ;CONTINUE IF RESET

 036740  013737  002420  002730          MOV     NA.IS,FP.NAM            ;LOAD REGISTER NAME ADDRESS
 036746  013737  002450  002746          MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
 036754                                  ERRHRD  3,M.SW4,ER.REG          ;"SW4 DOES NOT INIHIBIT AT0 BIT SET"
 036754  104456                          TRAP    C$ERHRD
 036756  000003                          .WORD   3
 036760  012153                          .WORD   M.SW4
 036762  015016                          .WORD   ER.REG
 036764  000410                          BR      400$                    ;EXIT TEST

 036766  023737  002736  002734  300$:   CMP     FP.ACT,FP.EXP           ;COMPARE VALUE
 036774  001404                          BEQ     400$                    ;IDENTICAL:  EXIT TEST

 036776                                  ERRHRD  4,M.ISR1,ER.DMP         ;"INTERFACE STATUS ERROR"
 036776  104456                          TRAP    C$ERHRD
 037000  000004                          .WORD   4
 037002  011543                          .WORD   M.ISR1
 037004  015074                          .WORD   ER.DMP

 037006  004737  015550          400$:   JSR     PC,CLRTST               ;AND CLEAR TEST DATA

 037012                                  ENDTST
 037012                          L10135:
 037012  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 23:  VERIFY SW5 DISABLES INTERFACE AT03 BIT
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      VERIFY SW5 IN "OFF" POSITION WILL DISABLE
                                ;;      USER ATTENTION BIT AT03 FROM ASSERTING ATA
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT REVA AND AT03 DISABLED THEN:
                                ;;          CLEAR MBC AND CHECK FOR ERROR
                                ;;          ISSUE INTERFACE CLEAR CMD AND TEST FOR ERROR STATUS
                                ;;          SET FS REGISTER FUNCTION BIT F3
                                ;;          READ IS REGISTER AND CHECK AT03 BIT SET & ATA BIT RESET
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      IS REGISTER AT03 & ATA BITS NOT SET/RESET PROPERLY
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T23::

                        000035  SELMASK =       B.UPAR + B.BYTE + B.ABORT + B.AT0
                        000002  SELTEST =       B.AT3

037014  013700  003010                MOV     SELECT,R0               ;READ SELECT WORD
037020  042700  000035                BIC     #SELMASK,R0             ;CLEAR DON'T CARE SELECT BITS
037024  022700  000002                CMP     #SELTEST,R0             ;COMPARE AND
037030  001402                        BEQ     50$                     ;BRANCH IF EQUAL

037032                                EXIT    TST                     ;EXIT TEST
037032  104432                        TRAP    C$EXIT
037034  000164                        .WORD   L10136-.

037036  004737  015342        50$:    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
037042  005737  003030                TST     MBCFLG                  ;TEST FLAG FOR ERROR
037046  001405                        BEQ     100$                    ;CONTINUE IF NO ERROR

037050                                ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
037050  104455                        TRAP    C$ERDF
037052  000001                        .WORD   1
037054  011650                        .WORD   M.MBC1
037056  015074                        .WORD   ER.DMP
037060                                DOCLN                           ;CLEAN UP AND EXIT
037060  104444                        TRAP    C$DCLN
```

TEST 23: VERIFY SW5 DISABLES INTERFACE AT03 BIT

```
7181
7182 037062  004737  015464          100$:   JSR     PC,CLRDRI               ;CLEAR DRIVE AND TEST FOR ERROR
7183 037066  005737  003032                  TST     DRIFLG                  ;TEST ERROR FLAG
7184 037072  001405                          BEQ     150$                    ;CONTINUE IF NO ERROR
7185
7186 037074                                  ERRHRD  2,M.ISR2,ER.DMP         ;"INTERFACE STATUS ERROR ON CLEAR CMD"
     037074  104456                          TRAP    C$ERHRD
     037076  000002                          .WORD   2
     037100  011572                          .WORD   M.ISR2
     037102  015074                          .WORD   ER.DMP
7187 037104  000443                          BR      400$                    ;EXIT TEST
7188
7189 037106  012777  004000  143170  150$:   MOV     #FS.F3,@A.FS            ;SET FUNCTION BIT F3
7190 037114  017737  143170  002736          MOV     @A.IS,FP.ACT            ;READ REGISTER
7191 037122  012777  000000  143154          MOV     #0,@A.FS                ;CLEAR FS REGISTER
7192
7193 037130  012737  000210  002734  200$:   MOV     #210,FP.EXP             ;LOAD EXPECTED:  IRY & AT3
7194 037136  032737  100000  002736          BIT     #IS.ATA,FP.ACT          ;TEST FOR ATA RESET
7195 037144  001413                          BEQ     300$                    ;CONTINUE IF RESET
7196
7197 037146  013737  002420  002730          MOV     NA.IS,FP.NAM            ;LOAD REGISTER NAME ADDRESS
7198 037154  013737  002450  002746          MOV     BA.IS,FP.TBL            ;LOAD REGISTER EXPAND TABLE ADDRESS
7199 037162                                  ERRHRD  3,M.SW5,ER.REG          ;"SW5 DOES NOT INIHIBIT AT3 BIT SET"
     037162  104456                          TRAP    C$ERHRD
     037164  000003                          .WORD   3
     037166  012233                          .WORD   M.SW5
     037170  015016                          .WORD   ER.REG
7200 037172  000410                          BR      400$                    ;EXIT TEST
7201
7202 037174  023737  002736  002734  300$:   CMP     FP.ACT,FP.EXP           ;COMPARE VALUE
7203 037202  001404                          BEQ     400$                    ;IDENTICAL:  EXIT TEST
7204
7205 037204                                  ERRHRD  4,M.ISR1,ER.DMP         ;"INTERFACE STATUS ERROR"
     037204  104456                          TRAP    C$ERHRD
     037206  000004                          .WORD   4
     037210  011543                          .WORD   M.ISR1
     037212  015074                          .WORD   ER.DMP
7206
7207 037214  004737  015550          400$:   JSR     PC,CLRTST               ;AND CLEAR TEST DATA
7208
7209 037220                                  ENDTST
     037220                          L10136:
     037220  104401                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 24:  VERIFY SW6 DISABLES INTERFACE ABORT LOGIC
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      VERIFY SW6 IN "OFF" POSITION WILL DISABLE
                                ;;      USER ATTENTION TRANSFER ABORT LOGIC
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF AT03 ABORT DISABLED AND NOT REVA THEN:
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                ;;          ISSUE INTERFACE WRITE W/INT ENABLE & SET CYCLE REQUEST
                                ;;          SET FS REGISTER FUNCTION BIT F3 AND WAIT FOR INTERRUPT
                                ;;          READ REGISTERS, CHECK IRY SET AND WORD COUNT IS ZERO
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      CONTROLLER NOT CLEARED BY MASSBUS INIT
                                ;;      INTERFACE STATUS ERROR
                                ;;      DATA LATE ERROR; ADJUST BANDWIDTH
                                ;;      SW1-6 DOES NOT DISABLE AT3 TRANSFER ABORT LOGIC
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T24::

                                SELMASK =       B.UPAR + B.BYTE + B.AT3 + B.AT0
                                SELTEST =       B.ABORT

037222  013700  003010                MOV     SELECT,R0               ;READ SELECT WORD
037226  042700  000033                BIC     #SELMASK,R0             ;MASK DON'T CARE SELECT BITS
037232  022700  000004                CMP     #SELTEST,R0             ;COMPARE AND
037236  001402                       BEQ     50$                     ;BRANCH IF EQUAL

037240                               EXIT    TST                     ;EXIT TEST
037240  104432                       TRAP    C$EXIT
037242  000244                       .WORD   L10137-.

037244  004737  015274       50$:    JSR     PC,SETISR               ;SETUP ISR VECTOR

037250  004737  015342               JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
037254  005737  003030               TST     MBCFLG                  ;TEST FLAG FOR ERROR
037260  001405                       BEQ     100$                    ;CONTINUE IF NO ERROR

037262                               ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
037262  104455                       TRAP    C$ERDF
037264  000001                       .WORD   1
037266  011650                       .WORD   M.MBC1
037270  015074                       .WORD   ER.DMP
037272                               DOCLN                           ;CLEAN UP AND EXIT
037272  104444                       TRAP    C$DCLN
```

```
 7256
 7257 037274  012700  177750          100$:  MOV     #-30,R0              ;LOAD BLOCK SIZE OF 24 WORDS (RH70)
 7258 037300  005737  003012                 TST     CPUFLG               ;TEST FOR PDP-11/70
 7259 037304  001002                         BNE     120$                 ;IF NE THEN YES
 7260 037306  012700  177640                 MOV     #-140,R0             ;LOAD BLOCK SIZE OF 96 WORDS (RH11)
 7261
 7262 037312  004737  015762          120$:  JSR     PC,WRTINI            ;AND INITIALIZE ADDRESS REGISTERS
 7263 037316  012777  000161  142752         MOV     #WIE.CMD,@A.CS1      ;ISSUE WRITE COMMAND TO CS1
 7264 037324  004737  016030                 JSR     PC,SILODLY           ;WAIT FOR SILO TO FILL
 7265 037330  012777  000020  142752         MOV     #IS.CYC,@A.IS        ;SET CYCLE REQUEST
 7266 037336  000240                         NOP                          ;DELAY JUST
 7267 037340  000240                         NOP                          ;A FEW CYCLES
 7268 037342  012777  004000  142734         MOV     #FS.F3,@A.FS         ;SET FS REGISTER FUNCTION BIT F3
 7269
 7270 037350  004737  015444          200$:  JSR     PC,TSTINT            ;WAIT FOR INTERRUPT OR TIME OUT
 7271 037354  004737  015606                 JSR     PC,RDREG             ;READ REGISTERS
 7272 037360  012777  000000  142716         MOV     #0,@A.FS             ;CLEAR FUNCTION / STATUS BITS
 7273
 7274 037366  122737  000210  002340         CMPB    #210,R.IS            ;TEST ISR
 7275 037374  001420                         BEQ     300$                 ;IF EQ THEN OK
 7276
 7277 037376  013737  002340  002736         MOV     R.IS,FP.ACT          ;LOAD ACTUAL VALUE
 7278 037404  012737  100210  002734         MOV     #100210,FP.EXP       ;EXPECTED:  ATA,IRY & AT3
 7279 037412  013737  002420  002730         MOV     NA.IS,FP.NAM         ;LOAD REGISTER NAME ADDRESS
 7280 037420  013737  002450  002746         MOV     BA.IS,FP.TBL         ;LOAD REGISTER EXPAND TABLE ADDRESS
 7281 037426                                 ERRHRD  2,M.ISR1,ER.REG      ;''INTERFACE STATUS ERROR''
      037426  104456                        TRAP    C$ERHRD
      037430  000002                        .WORD   2
      037432  011543                        .WORD   M.ISR1
      037434  015016                        .WORD   ER.REG
 7282
 7283 037436  032737  100000  002336  300$:  BIT     #CS2.DLT,R.CS2       ;TEST FOR DATA LATE
 7284 037444  001405                         BEQ     400$                 ;IF EQ THEN OK
 7285
 7286 037446                                 ERRHRD  3,M.DLT1,ER.DMP      ;''DATA LATE ERROR; ADJUST BANDWIDTH''
      037446  104456                        TRAP    C$ERHRD
      037450  000003                        .WORD   3
      037452  007707                        .WORD   M.DLT1
      037454  015074                        .WORD   ER.DMP
 7287 037456  000407                         BR      500$                 ;SKIP WORD COUNT CHECK
 7288
 7289 037460  005737  002330          400$:  TST     R.WC                 ;TEST WORD COUNT
 7290 037464  001404                         BEQ     500$                 ;IF WORD COUNT IS ZERO THEN OK
 7291
 7292 037466                                 ERRHRD  4,M.SW6,ER.DMP       ;''SW1-6 DOES NOT DISABLE XFER ABORT''
      037466  104456                        TRAP    C$ERHRD
      037470  000004                        .WORD   4
      037472  012313                        .WORD   M.SW6
      037474  015074                        .WORD   ER.DMP
 7293
 7294 037476  004737  015324          500$:  JSR     PC,CLRISR            ;CLEAR ISR VECTOR
 7295
 7296 037502  004737  015550                 JSR     PC,CLRTST            ;AND CLEAR TEST DATA
 7297
 7298 037506                                 ENDTST
      037506                         L10137:
      037506  104401                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 25:  USER DATA WORD BLOCK TRANSFER
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      PERFORM 64-WORD DMA DATA TRANSFERS, WITH USER SPECIFIED PATTERN
                                ;;      COMPARE DATA READ AGAINST DATA WRITTEN
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF NOT BYTE MODE AND DPFLG THEN:
                                ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                ;;          GET DATA PATTERN FROM OPERATOR, AND SETUP BUFFER PATTERN
                                ;;          EXECUTE SUBTEST 1 AND CLEAR INTERRUPT VECTOR
                                ;;          TEST FOR OPERATOR CONTROL/C TO EXIT
                                ;;
                                ;;    ERROR CONDITIONS:
                                ;;
                                ;;      MASSBUS CONTROLLER ERROR STATUS
                                ;;      DATA TRANSFER NOT COMPLETE
                                ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                ;;      DATA RECEIVED DOES NOT MATCH EXPECTED DATA
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T25::

                        000010  SELTEST =       B.BYTE

037510  005737  002242                TST     DPFLG                   ;TEST DP FLAG
037514  001002                        BNE     25$                     ;BRANCH IF SET

037516                          EXIT    TST                             ;FLAG NOT SET:  EXIT TEST
037516  104432                        TRAP    C$EXIT
037520  000514                        .WORD   L10140-.

037522  013700  003010        25$:    MOV     SELECT,R0               ;READ SELECT WORD
037526  032700  000010                BIT     #SELTEST,R0             ;AND TEST BYTE MODE BIT SET
037532  001402                        BEQ     50$                     ;BRANCH IF NOT SET

037534                          EXIT    TST                             ;BYTE MODE:  EXIT TEST
037534  104432                        TRAP    C$EXIT
037536  000476                        .WORD   L10140-.

037540  004737  015274        50$:    JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
037544  012737  000000  002514        MOV     #0,D.WDP                ;CLEAR USER DATA PATTERN
037552  012737  000100  002506        MOV     #100,D.BLOCK            ;INITIALIZE BLOCK SIZE
037560  012737  000200  002512        MOV     #200,D.MAX              ;INITIALIZE MAX DATA BUFFER INDEX

037566                  100$:   GMANID  Q.WDP,D.WDP,0,-1,0,177777,YES
037566  104443                  TRAP    C$GMAN
037570  000406                  BR      10000$
037572  002514                  .WORD   D.WDP
037574  000032                  .WORD   T$CODE
037576  014220                  .WORD   Q.WDP
```

```
        037600  177777                          .WORD   -1
        037602  000000                          .WORD   T$LOLIM
        037604  177777                          .WORD   T$HILIM
        037606                          10000$:
  7345
  7346  037606                          200$:   GMANID  Q.WBL,D.BLOCK,D,-1,1,100,YES
        037606  104443                          TRAP    C$GMAN
        037610  000406                          BR      10001$
        037612  002506                          .WORD   D.BLOCK
        037614  000052                          .WORD   T$CODE
        037616  014274                          .WORD   Q.WBL
        037620  177777                          .WORD   -1
        037622  000001                          .WORD   T$LOLIM
        037624  000100                          .WORD   T$HILIM
        037626                          10001$:
  7347
  7348  037626  013701  002506                  MOV     D.BLOCK,R1              ;LOAD USER BLOCK SIZE
  7349  037632  006301                          ASL     R1                      ;DOUBLE FOR ADDRESS INDEX
  7350  037634  010137  002512                  MOV     R1,D.MAX                ;AND SAVE AS MAX INDEX
  7351  037640  005437  002506                  NEG     D.BLOCK                 ;COMPLEMENT
  7352
  7353  037644  005002                  LP.WDP: CLR     R2                      ;WORD PATTERN LOOP LABEL
  7354  037646  013701  002514                  MOV     D.WDP,R1                ;LOAD DATA PATTERN
  7355  037652  010137  002744                  MOV     R1,FP.DPAT              ;INITIALIZE DISPLAY PARM
  7356
  7357  037656  010162  004054          250$:   MOV     R1,OBUF(R2)             ;LOAD USER PATTERN INTO BUFFER
  7358  037662  012762  000000  003054          MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
  7359  037670  062702  000002                  ADD     #2,R2                   ;AND BUMP INDEX
  7360  037674  023702  002512                  CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
  7361  037700  003366                          BGT     250$                    ;CONTINUE UNTIL BUFFER INITIALIZED
```

```
 7363 037702                            STARS
                                        ;;*****************************************************************
 7364                                   ;;
 7365                                   ;;    SUBTEST 1:
 7366                                   ;;
 7367                                   ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 7368                                   ;;
 7369                                   ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 7370                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 7371                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 7372                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 7373                                   ;;
 7374                                   ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 7375                                   ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 7376                                   ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 7377                                   ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 7378                                   ;;
 7379                                   ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 7380 037702                            STARS
                                        ;;*****************************************************************
 7381
 7382 037702                                    BGNSUB
      037702                            T25.1:
      037702  104402                            TRAP    C$BSUB
 7383
 7384 037704  004737  015342                    JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 7385 037710  005737  003030                    TST     MBCFLG                  ;TEST FLAG FOR ERROR
 7386 037714  001405                            BEQ     100$                    ;CONTINUE IF NO ERROR
 7387
 7388 037716                                    ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      037716  104455                            TRAP    C$ERDF
      037720  000001                            .WORD   1
      037722  011650                            .WORD   M.MBC1
      037724  015074                            .WORD   ER.DMP
 7389 037726                                    DOCLN                           ;CLEAN UP AND EXIT
      037726  104444                            TRAP    C$DCLN
 7390
 7391 037730  005002                    100$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 7392
 7393 037732  013700  002506            200$:   MOV     D.BLOCK,R0              ;LOAD BLOCK SIZE
 7394 037736  004737  015674                    JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 7395 037742  004737  015410                    JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 7396 037746  005737  003022                    TST     INTFLG                  ;TEST FOR INTERRUPT
 7397 037752  001005                            BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 7398
 7399 037754                                    ERRHRD  2,M.INT1,ER.DMP         ;'NOT INTERRUPT ON WORD XFER EBL"
      037754  104456                            TRAP    C$ERHRD
      037756  000002                            .WORD   2
      037760  010706                            .WORD   M.INT1
      037762  015074                            .WORD   ER.DMP
 7400 037764  000407                            BR      300$                    ;EXIT SEGMENT
 7401
 7402 037766  005737  003030            250$:   TST     MBCFLG                  ;TEST FOR ERROR
 7403 037772  001404                            BEQ     300$                    ;CONTINUE IF RESET
 7404
 7405 037774                                    ERRHRD  3,M.WRD2,ER.DMP         ;'WORD XFER ERROR ON INTERFACE WRITE"
      037774  104456                            TRAP    C$ERHRD
```

```
        037776  000003                          .WORD   3
        040000  007324                          .WORD   M.WRD2
        040002  015074                          .WORD   ER.DMP
   7406
   7407 040004  013700  002506          300$:   MOV     D.BLOCK,R0              ;LOAD BLOCK SIZE
   7408 040010  004737  015642                  JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
   7409 040014  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
   7410 040020  005737  003022                  TST     INTFLG                  ;TEST FOR INTERRUPT
   7411 040024  001005                          BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
   7412
   7413 040026                                  ERRHRD  4,M.INT1,ER.DMP         ;'NOT INTERRUPT ON WORD XFER EBL''
        040026  104456                          TRAP    C$ERHRD
        040030  000004                          .WORD   4
        040032  010706                          .WORD   M.INT1
        040034  015074                          .WORD   ER.DMP
   7414 040036  000407                          BR      400$                    ;EXIT SEGMENT
   7415
   7416 040040  005737  003030          350$:   TST     MBCFLG                  ;TEST FOR ERROR
   7417 040044  001404                          BEQ     400$                    ;CONTINUE IF RESET
   7418
   7419 040046                                  ERRHRD  5,M.WRD3,ER.DMP         ;'WORD XFER ERROR ON INTERFACE READ''
        040046  104456                          TRAP    C$ERHRD
        040050  000005                          .WORD   5
        040052  007373                          .WORD   M.WRD3
        040054  015074                          .WORD   ER.DMP
```

```
7421
7422 040056  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
7423 040060  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
7424
7425 040064  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
7426 040072  001427                            BEQ     475$                    ;IDENTICAL:  CONTINUE
7427
7428 040074  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
7429 040100  001004                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
7430
7431 040102                                    ERRHRD  6,M.WRD4,ER.DATA        ;'DATA COMPARISON ERROR''
     040102  104456                            TRAP    C$ERHRD
     040104  000006                            .WORD   6
     040106  007441                            .WORD   M.WRD4
     040110  015172                            .WORD   ER.DATA
7432
7433 040112  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
7434 040116  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
7435 040124  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
7436
7437 040126  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
7438 040132  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
7439 040140  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
7440 040146  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
7441
7442 040152  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
7443 040156  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
7444 040162  003340                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
7445
7446 040164  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
7447 040170  001414                            BEQ     600$                    ;BRANCH IF NO ERRORS
7448
7449 040172                                    PRINTX  #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     040172  013746  003026                    MOV     ERRFLG,-(SP)
     040176  012746  013304                    MOV     #F.ERCNT,-(SP)
     040202  012746  000002                    MOV     #2,-(SP)
     040206  010600                            MOV     SP,R0
     040210  104415                            TRAP    C$PNTX
     040212  062706  000006                    ADD     #6,SP
7450 040216  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
7451
7452 040222                            600$:   ENDSUB
     040222                         L10141:
     040222  104403                            TRAP    C$ESUB
7453
7454 040224                                    BREAK                           ;TEST FOR CONTROL/C
     040224  104422                            TRAP    C$BRK
7455
7456 040226  000606                            BR      LP.WDP                  ;LOOP BACK
7457
7458 040230  004737  015550            700$:   JSR     PC,CLRTST               ;AND CLEAR TEST DATA
7459
7460 040234                                    ENDTST                          ;END OF TEST
     040234                         L10140:
     040234  104401                            TRAP    C$ETST
```

```
                                          .SBTTL  TEST 26:  USER DATA BYTE BLOCK TRANSFER
     040236                               STARS
                                          ;;*****************************************************************
                                          ;;
                                          ;;  TEST DESCRIPTION:
                                          ;;
                                          ;;      PERFORM 64-BYTE DMA DATA TRANSFERS, WITH USER SPECIFIED PATTERN
                                          ;;      COMPARE DATA READ AGAINST DATA WRITTEN
                                          ;;
                                          ;;    TEST STEPS:
                                          ;;
                                          ;;      IF BYTE MODE AND DPFLG THEN:
                                          ;;          SETUP INTERRUPT SERVICE ROUTINE VECTOR
                                          ;;          GET DATA PATTERN FROM OPERATOR, AND SETUP BUFFER PATTERN
                                          ;;          EXECUTE SUBTEST 1 AND CLEAR INTERRUPT VECTOR
                                          ;;          TEST FOR OPERATOR CONTROL/C TO EXIT
                                          ;;
                                          ;;    ERROR CONDITIONS:
                                          ;;
                                          ;;      MASSBUS CONTROLLER ERROR STATUS
                                          ;;      DATA TRANSFER NOT COMPLETE
                                          ;;      NO INTERRUPT ON DATA TRANSFER COMPLETE
                                          ;;      DATA RECEIVED DOES NOT MATCH EXPECTED DATA
     040236                               STARS
                                          ;;*****************************************************************

     040236                                       BGNTST
     040236                               T26::

             000010                       SELTEST =       B.BYTE

     040236  005737  002242                       TST     DPFLG                   ;TEST DP FLAG
     040242  001002                               BNE     25$                     ;BRANCH IF SET

     040244                                       EXIT    TST                     ;FLAG NOT SET:  EXIT TEST
     040244  104432                               TRAP    C$EXIT
     040246  000536                               .WORD   L10142-.

     040250  013700  003010               25$:    MOV     SELECT,R0               ;READ SELECT WORD
     040254  032700  000010                       BIT     #SELTEST,R0             ;AND TEST BYTE MODE BIT SET
     040260  001002                               BNE     50$                     ;BRANCH IF SET

     040262                                       EXIT    TST                     ;BYTE MODE:  EXIT TEST
     040262  104432                               TRAP    C$EXIT
     040264  000520                               .WORD   L10142-.

     040266  004737  015274               50$:    JSR     PC,SETISR               ;SETUP INTERRUPT SERVICE VECTOR
     040272  012737  000000  002516               MOV     #0,D.BDP                ;RESET USER DATA PATTERN
     040300  012737  000100  002506               MOV     #100,D.BLOCK            ;INITIALIZE BLOCK SIZE
     040306  012737  000100  002512               MOV     #100,D.MAX              ;INITIALIZE MAX DATA BUFFER INDEX

     040314                               100$:   CMANID  Q.BDP,D.BDP,0,-1,0,377,YES
     040314  104443                               TRAP    C$GMAN
     040316  000406                               BR      10000$
     040320  002516                               .WORD   D.BDP
     040322  000032                               .WORD   T$CODE
     040324  014246                               .WORD   Q.BDP
```

```
        040326  177777                          .WORD   -1
        040330  000000                          .WORD   T$LOLIM
        040332  000377                          .WORD   T$HILIM
        040334                          10000$:
  7507
  7508 040334  005001                           CLR     R1                      ;CLEAR REGISTER
  7509 040336  113701  002516                   MOVB    D.BDP,R1                ;LOAD BYTE PATTERN
  7510 040342  000301                           SWAB    R1                      ;SWAP BYTES
  7511 040344  050137  002516                   BIS     R1,D.BDP                ;SETUP HIGH BYTE
  7512
  7513 040350                          200$:    GMANID  Q.BBL,D.BLOCK,D,-1,10,100,YES
        040350  104443                          TRAP    C$GMAN
        040352  000406                          BR      10001$
        040354  002506                          .WORD   D.BLOCK
        040356  000052                          .WORD   T$CODE
        040360  014314                          .WORD   Q.BBL
        040362  177777                          .WORD   -1
        040364  000010                          .WORD   T$LOLIM
        040366  000100                          .WORD   T$HILIM
        040370                          10001$:
  7514
  7515 040370  013701  002506                   MOV     D.BLOCK,R1              ;LOAD USER BLOCK SIZE
  7516 040374  042701  000001                   BIC     #1,R1                   ;INSURE NUMBER OF BYTES IS EVEN
  7517 040400  010137  002512                   MOV     R1,D.MAX                ;SAVE AS MAX INDEX
  7518 040404  006201                           ASR     R1                      ;DIVIDE BY TWO
  7519 040406  005401                           NEG     R1                      ;COMPLEMENT
  7520 040410  010137  002506                   MOV     R1,D.BLOCK              ;AND RESTORE
  7521
  7522 040414  005002                  LP.BDP:  CLR     R2                      ;WORD PATTERN LOOP LABEL
  7523 040416  013701  002516                   MOV     D.BDP,R1                ;LOAD DATA PATTERN
  7524 040422  010137  002744                   MOV     R1,FP.DPAT              ;INITIALIZE DISPLAY PATTERN
  7525
  7526 040426  010162  004054           250$:   MOV     R1,OBUF(R2)             ;LOAD USER PATTERN INTO BUFFER
  7527 040432  012762  000000  003054           MOV     #0,IBUF(R2)             ;CLEAR INPUT BUFFER
  7528 040440  062702  000002                   ADD     #2,R2                   ;AND BUMP INDEX
  7529 040444  023702  002512                   CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
  7530 040450  003366                           BGT     250$                    ;CONTINUE UNTIL BUFFER INITIALIZED
```

```
 7532 040452                                STARS
                                            ;;*****************************************************************
 7533                                       ;;
 7534                                       ;;    SUBTEST 1:
 7535                                       ;;
 7536                                       ;;      CLEAR THE MASSBUS CONTROLLER AND CHECK FOR ERROR
 7537                                       ;;
 7538                                       ;;      INITIALIZE REGISTERS FOR WRITE W/INT ENABLE
 7539                                       ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 7540                                       ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 7541                                       ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 7542                                       ;;
 7543                                       ;;      INITIALIZE REGISTERS FOR READ W/INT ENABLE
 7544                                       ;;      SET CYCLE REQUEST, DELAY AND READ REGISTERS
 7545                                       ;;      CHECK FOR ERROR STATUS AND CHECK FOR INTERRUPT
 7546                                       ;;      IF ERROR OR NO INTERRUPT THEN:  EXIT SUBTEST
 7547                                       ;;
 7548                                       ;;      COMPARE DATA READ AGAINST DATA WRITTEN
 7549 040452                                STARS
                                            ;;*****************************************************************
 7550
 7551 040452                                        BGNSUB
      040452                                T26.1:
      040452  104402                               TRAP    C$BSUB
 7552
 7553 040454  004737  015342                       JSR     PC,CLRMBC               ;CLEAR MASSBUS CONTROLLER
 7554 040460  005737  003030                       TST     MBCFLG                  ;TEST FLAG FOR ERROR
 7555 040464  001405                               BEQ     100$                    ;CONTINUE IF NO ERROR
 7556
 7557 040466                                       ERRDF   1,M.MBC1,ER.DMP         ;DEVICE FATAL
      040466  104455                               TRAP    C$ERDF
      040470  000001                               .WORD   1
      040472  011650                               .WORD   M.MBC1
      040474  015074                               .WORD   ER.DMP
 7558 040476                                       DOCLN                           ;CLEAN UP AND EXIT
      040476  104444                               TRAP    C$DCLN
 7559
 7560 040500  005002                        100$:  CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
 7561
 7562 040502  013700  002506                200$:  MOV     D.BLOCK,R0              ;LOAD BLOCK SIZE
 7563 040506  004737  015674                       JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
 7564 040512  004737  015410                       JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
 7565 040516  005737  003022                       TST     INTFLG                  ;TEST FOR INTERRUPT
 7566 040522  001005                               BNE     250$                    ;BRANCH IF INTERRUPT RECEIVED
 7567
 7568 040524                                       ERRHRD  2,M.INT1,ER.DMP         ;'NOT INTERRUPT ON WORD XFER EBL"
      040524  104456                               TRAP    C$ERHRD
      040526  000002                               .WORD   2
      040530  010706                               .WORD   M.INT1
      040532  015074                               .WORD   ER.DMP
 7569 040534  000407                               BR      300$                    ;EXIT SEGMENT
 7570
 7571 040536  005737  003030                250$:  TST     MBCFLG                  ;TEST FOR ERROR
 7572 040542  001404                               BEQ     300$                    ;CONTINUE IF RESET
 7573
 7574 040544                                       ERRHRD  3,M.BYT2,ER.DMP         ;'WORD XFER ERROR ON INTERFACE WRITE"
      040544  104456                               TRAP    C$ERHRD
```

```
        040546  000003                          .WORD   3
        040550  007533                          .WORD   M.BYT2
        040552  015074                          .WORD   ER.DMP
   7575
   7576 040554  013700  002506          300$:  MOV     D.BLOCK,R0              ;LOAD BLOCK SIZE
   7577 040560  004737  015642                 JSR     PC,RDBLK                ;READ BLOCK FROM INTERFACE
   7578 040564  004737  015410                 JSR     PC,TSTMBC               ;READ REGISTERS AND TEST ERRORS
   7579 040570  005737  003022                 TST     INTFLG                  ;TEST FOR INTERRUPT
   7580 040574  001005                         BNE     350$                    ;BRANCH IF INTERRUPT RECEIVED
   7581
   7582 040576                                 ERRHRD  4,M.INT1,ER.DMP         ;'NOT INTERRUPT ON WORD XFER EBL''
        040576  104456                         TRAP    C$ERHRD
        040600  000004                         .WORD   4
        040602  010706                         .WORD   M.INT1
        040604  015074                         .WORD   ER.DMP
   7583 040606  000407                         BR      400$                    ;EXIT SEGMENT
   7584
   7585 040610  005737  003030          350$:  TST     MBCFLG                  ;TEST FOR ERROR
   7586 040614  001404                         BEQ     400$                    ;CONTINUE IF RESET
   7587
   7588 040616                                 ERRHRD  5,M.BYT3,ER.DMP         ;'WORD XFER ERROR ON INTERFACE READ''
        040616  104456                         TRAP    C$ERHRD
        040620  000005                         .WORD   5
        040622  007602                         .WORD   M.BYT3
        040624  015074                         .WORD   ER.DMP
```

```
7590
7591 040626  005002                    400$:   CLR     R2                      ;CLEAR BUFFER INDEX REGISTER
7592 040630  005037  003026                    CLR     ERRFLG                  ;CLEAR ERROR COUNT
7593
7594 040634  026262  003054  004054    425$:   CMP     IBUF(R2),OBUF(R2)       ;COMPARE DATA
7595 040642  001427                            BEQ     475$                    ;IDENTICAL:  CONTINUE
7596
7597 040644  005737  003026                    TST     ERRFLG                  ;TEST ERROR COUNT
7598 040650  001004                            BNE     450$                    ;BRANCH IF NOT FIRST ERROR
7599
7600 040652                                    ERRHRD  6,M.BYT4,ER.DATA        ;''DATA COMPARISON ERROR''
     040652  104456                            TRAP    C$ERHRD
     040654  000006                            .WORD   6
     040656  007650                            .WORD   M.BYT4
     040660  015172                            .WORD   ER.DATA
7601
7602 040662  005237  003026            450$:   INC     ERRFLG                  ;BUMP ERROR COUNT
7603 040666  022737  000010  003026            CMP     #8.,ERRFLG              ;COMPARE WITH MAX ERROR COUNT
7604 040674  002412                            BLT     475$                    ;BRANCH IF MAX LESS THAN COUNT
7605
7606 040676  010237  002742                    MOV     R2,FP.NDX               ;SAVE DATA INDEX
7607 040702  016237  003054  002736            MOV     IBUF(R2),FP.ACT         ;SAVE RECEIVED DATA
7608 040710  016237  004054  002734            MOV     OBUF(R2),FP.EXP         ;SAVE DATA SENT
7609 040716  004737  016374                    JSR     PC,DSPDATA              ;AND DISPLAY DATA
7610
7611 040722  062702  000002            475$:   ADD     #2,R2                   ;BUMP INDEX
7612 040726  023702  002512                    CMP     D.MAX,R2                ;COMPARE WITH MAX BUFFER INDEX
7613 040732  003340                            BGT     425$                    ;CONTINUE UNTIL INDEX EQUALS MAX
7614
7615 040734  005737  003026            500$:   TST     ERRFLG                  ;TEST NUMBER ERRORS
7616 040740  001414                            BEQ     600$                    ;BRANCH IF NO ERRORS
7617
7618 040742                                    PR.INTX #F.ERCNT,ERRFLG         ;PRINT SUBTEST ERROR COUNT
     040742  013746  003026                    MOV     ERRFLG,-(SP)
     040746  012746  013304                    MOV     #F.ERCNT,-(SP)
     040752  012746  000002                    MOV     #2,-(SP)
     040756  010600                            MOV     SP,R0
     040760  104415                            TRAP    C$PNTX
     040762  062706  000006                    ADD     #6,SP
7619 040766  005237  003004                    INC     ERRCNT                  ;ADD TO TEST ERROR COUNT
7620
7621 040772                            600$:   ENDSUB
     040772                            L10143:
     040772  104403                            TRAP    C$ESUB
7622
7623 040774                                    BREAK                           ;TEST FOR CONTROL/C
     040774  104422                            TRAP    C$BRK
7624
7625 040776  000606                            BR      LP.BDP                  ;LOOP BACK
7626
7627 041000  004737  015550            700$:   JSR     PC,CLRTST               ;AND CLEAR TEST DATA
7628
7629 041004                                    ENDTST                          ;END OF TEST
     041004                            L10142:
     041004  104401                            TRAP    C$ETST
```

```
                                .SBTTL  TEST 27:  ADJUST TRANSFER BANDWIDTH
                                STARS
                                ;;*****************************************************************
                                ;;
                                ;;  TEST DESCRIPTION:
                                ;;
                                ;;      MANUAL TEST FOR ADJUSTING SCLK PERIOD
                                ;;      LOOP ON 64 WORD WRITE
                                ;;
                                ;;    TEST STEPS:
                                ;;
                                ;;      IF BWFLG SET THEN:  REPEAT
                                ;;          CLEAR MASSBUS CONTROLLER AND CHECK FOR ERROR
                                ;;          INITIALIZE REGISTERS FOR READ W/INT ENABLE
                                ;;          SET CYCLE REQUEST, DELAY AND READ REGISTERS
                                ;;
                                STARS
                                ;;*****************************************************************

                                        BGNTST
                                T27::

041006  005737  002244                  TST     BWFLG           ;TEST BW FLAG
041012  001002                          BNE     50$             ;BRANCH IF SET

041014                                  EXIT    TST             ;FLAG NOT SET:  EXIT TEST
041014  104432                          TRAP    C$EXIT
041016  000110                          .WORD   L10144-.

041020  004737  015342          50$:    JSR     PC,CLRMBC       ;CLEAR MASSBUS CONTROLLER
041024  005737  003030                  TST     MBCFLG          ;TEST FLAG FOR ERROR
041030  001405                          BEQ     100$            ;CONTINUE IF NO ERROR

041032                                  ERRDF   1,M.MBC1,ER.DMP ;DEVICE FATAL
041032  104455                          TRAP    C$ERDF
041034  000001                          .WORD   1
041036  011650                          .WORD   M.MBC1
041040  015074                          .WORD   ER.DMP
041042                                  DOCLN                   ;CLEAN UP AND EXIT
041042  104444                          TRAP    C$DCLN

041044  012737  177700  002506  100$:   MOV     #-100,D.BLOCK   ;INITIALIZE FOR 64 WORD BLOCKS
041052  004737  015274                  JSR     PC,SETISR       ;SETUP ISR VECTOR

041056                                  PRINTB  #F.BW           ;PRINT MESSAGE AND BEGIN LOOP
041056  012746  013544                  MOV     #F.BW,-(SP)
041062  012746  000001                  MOV     #1,-(SP)
041066  010600                          MOV     SP,R0
041070  104414                          TRAP    C$PNTB
041072  062706  000004                  ADD     #4,SP
```

```
7666
7667 041076  013700  002506          200$:   MOV     D.BLOCK,R0              ;LOAD BLOCK SIZE
7668 041102  004737  015674                  JSR     PC,WRTBLK               ;WRITE BLOCK TO INTERFACE
7669 041106  004737  015410                  JSR     PC,TSTMBC               ;READ REGISTERS
7670
7671 041112                                  BREAK                           ;CHECK FOR OPERATOR CNTRL/C
     041112  104422                          TRAP    C$BRK
7672
7673 041114  000770                          BR      200$                    ;AND REPEAT
7674
7675 041116  004737  015324          400$:   JSR     PC,CLRISR               ;CLEAR VECTOR
7676
7677 041122  004737  015550                  JSR     PC,CLRTST               ;AND CLEAR TEST DATA
7678
7679 041126                                  ENDTST
     041126                          L10144:
     041126  104401                          TRAP    C$ETST
7680
7681 041130                                  ENDMOD
```

```
7683                                    .SBTTL
```

```
                                        .SBTTL  HARDWARE PARAMETER CODING SECTION

041130                                          BGNMOD

041130                                  STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      P-TABLE PARAMETER CODING:
                                        ;;
                                        ;;         OFFSET        DESCRIPTION
                                        ;;
                                        ;;              0        DEVICE CSR ADDRESS
                                        ;;              2        INTERRUPT VECTOR ADDRESS
                                        ;;              4        BUS REQUEST PRIORITY
                                        ;;              6        DRIVE/UNIT NUMBER
041130                                  STARS
                                        ;;*****************************************************************

041130                                          BGNHRD
041130  000022                                  .WORD L10145-L$HARD/2
041132                                  L$HARD::

041132                                          GPRMA   PQ.CSR,0,0,0,177776,YES         ;DEVICE BASE ADDRESS
041132  000031                                  .WORD   T$CODE
041134  041176                                  .WORD   PQ.CSR
041136  000000                                  .WORD   T$LOLIM
041140  177776                                  .WORD   T$HILIM

041142                                          GPRMA   PQ.VEC,2,0,0,776,YES            ;INTERRUPT VECTOR ADDRESS
041142  001031                                  .WORD   T$CODE
041144  041216                                  .WORD   PQ.VEC
041146  000000                                  .WORD   T$LOLIM
041150  000776                                  .WORD   T$HILIM

041152                                          GPRMD   PQ.PRI,4,D,-1,0,6,YES           ;BUS REQUEST PRIORITY
041152  002052                                  .WORD   T$CODE
041154  041236                                  .WORD   PQ.PRI
041156  177777                                  .WORD   -1
041160  000000                                  .WORD   T$LOLIM
041162  000006                                  .WORD   T$HILIM

041164                                          GPRMD   PQ.DRI,6,D,-1,0,7,YES           ;DRIVE SELECT
041164  003052                                  .WORD   T$CODE
041166  041256                                  .WORD   PQ.DRI
041170  177777                                  .WORD   -1
041172  000000                                  .WORD   T$LOLIM
041174  000007                                  .WORD   T$HILIM

041176                                          ENDHRD
                                                .EVEN
041176                                  L10145:
```

```
7754
7755 041176                            STARS
                                       ;;*****************************************************************
7756                                   ;;
7757                                   ;;      P-TABLE QUESTION STORAGE
7758 041176                            STARS
                                       ;;*****************************************************************
7759
7760                                           .NLIST  BEX
7761
7762 041176    104    105    126  PQ.CSR:      .ASCIZ  /DEVICE ADDRESS/
7763                                           .EVEN
7764 041216    126    105    103  PQ.VEC:      .ASCIZ  /VECTOR ADDRESS/
7765                                           .EVEN
7766 041236    120    122    111  PQ.PRI:      .ASCIZ  /PRIORITY      /
7767                                           .EVEN
7768 041256    104    122    111  PQ.DRI:      .ASCIZ  /DRIVE         /
7769                                           .EVEN
7770
7771                                           .LIST   BEX
```

```
 7773                                   .SBTTL          SOFTWARE P-TABLE PARAMETER CODING
 7774
 7775 041276                            STARS
                                        ;;*****************************************************************
                                        ;;
                                        ;;      SOFTWARE P-TABLE PARAMETER CODING:
                                        ;;
                                        ;;         OFFSET       DESCRIPTION
                                        ;;
                                        ;;              0       DR70 REV "A"
                                        ;;              2       SWITCH 4 AT0 ATTN DISABLE
                                        ;;              4       SWITCH 5 AT3 ATTN DISABLE
                                        ;;              6       SWITCH 6 AT3 ABORT LOGIC DISABLE
                                        ;;              8       SWITCH 7 BYTE MODE TRANSFER
                                        ;;              10      SWITCH 8 USER PARITY ENABLE
                                        ;;              12      ADJUST TRANSFER BANDWIDTH
 7788 041276                            STARS
                                        ;;*****************************************************************
 7789
 7790 041276                                    BGNSFT
      041276  000030                            .WORD L10146-L$SOFT/2
      041300                            L$SOFT::
 7791
 7792 041300                                    GPRML   SQ.REV,0,1,YES          ;ASK FOR REV INFO
      041300  000130                            .WORD   T$CODE
      041302  041360                            .WORD   SQ.REV
      041304  000001                            .WORD   1
 7793
 7794 041306                                    GPRML   SQ.SW4,2,1,YES          ;ASK FOR AT0 ATTN DISABLE
      041306  001130                            .WORD   T$CODE
      041310  041410                            .WORD   SQ.SW4
      041312  000001                            .WORD   1
 7795
 7796 041314                                    GPRML   SQ.SW5,4,1,YES          ;ASK FOR AT3 ATTN DISABLE
      041314  002130                            .WORD   T$CODE
      041316  041450                            .WORD   SQ.SW5
      041320  000001                            .WORD   1
 7797
 7798 041322                                    GPRML   SQ.SW6,6,1,YES          ;ASK FOR AT3 XFER ABORT DISABLE
      041322  003130                            .WORD   T$CODE
      041324  041510                            .WORD   SQ.SW6
      041326  000001                            .WORD   1
 7799
 7800 041330                                    GPRML   SQ.SW7,10,1,YES         ;ASK FOR BYTE MODE TRANSFER
      041330  004130                            .WORD   T$CODE
      041332  041562                            .WORD   SQ.SW7
      041334  000001                            .WORD   1
 7801
 7802 041336                                    GPRML   SQ.SW8,12,1,YES         ;ASK FOR USER PARITY ENABLE
      041336  005130                            .WORD   T$CODE
      041340  041622                            .WORD   SQ.SW8
      041342  000001                            .WORD   1
 7803
 7804 041344                                    GPRML   SQ.DP,14,1,YES          ;ASK IF USER DATA PATTERN TEST
      041344  006130                            .WORD   T$CODE
      041346  041666                            .WORD   SQ.DP
      041350  000001                            .WORD   1
```

```
ZDRMA0 DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 213-1
SOFTWARE P-TABLE PARAMETER CODING

   7805
   7806 041352                                  GPRML   SQ.BW,16,1,YES          ;ASK IF ADJUST TRANSFER BANDWIDTH
        041352  007130                          .WORD   T$CODE
        041354  041726                          .WORD   SQ.BW
        041356  000001                          .WORD   1
   7807
   7808 041360                                  ENDSFT
                                                .EVEN
        041360                          L10146:
   7809
```

```
7811
7812 041360                            STARS
                                       ;;*****************************************************************
7813                                   ;;
7814                                   ;;      SOFTWARE P-TABLE QUESTION STORAGE
7815 041360                            STARS
                                       ;;*****************************************************************
7816
7817                                           .NLIST  BEX
7818
7819 041360  122  105  126  SQ.REV:             .ASCIZ  /REVISION "A" INTERFACE/
7820                                           .EVEN
7821 041410  101  124  060  SQ.SW4:             .ASCIZ  /AT0 ATTN DISABLED  [SW4 'OFF']/
7822                                           .EVEN
7823 041450  101  124  063  SQ.SW5:             .ASCIZ  /AT3 ATTN DISABLED  [SW5 'OFF']/
7824                                           .EVEN
7825 041510  101  124  063  SQ.SW6:             .ASCIZ  /AT3 TRANSFER ABORT DISABLED  [SW6 'OFF']/
7826                                           .EVEN
7827 041562  102  131  124  SQ.SW7:             .ASCIZ  /BYTE MODE OPERATION  [SW7 'ON']/
7828                                           .EVEN
7829 041622  111  116  120  SQ.SW8:             .ASCIZ  /INPUT PARITY DISABLED  [SW8 'OFF']/
7830                                           .EVEN
7831 041666  117  120  105  SQ.DP:              .ASCIZ  /OPERATOR SPECIFIED DATA PATTERN/
7832                                           .EVEN
7833 041726  101  104  112  SQ.BW:              .ASCIZ  /ADJUST TRANSFER BANDWIDTH/
7834                                           .EVEN
7835
7836                                           .LIST   BEX
7837
```

```
   7839
   7840 041760                          $PATCH::
   7841
   7842 041760                                  .BLKW   40      ;32 WORD PATCH AREA
   7843
   7844 042060                                  LASTAD
                                                .EVEN
        042060  042100                         .WORD T$FREE
        042062  000006                         .WORD T$SIZE
        042064                          L$LAST::
   7845
```

```
                                .SBTTL  SETUP P-TABLE SECTION

042064                          STARS
                                ;;*****************************************************************
                                ;;
                                ;;      INITIALIZE P-TABLE FOR ONE UNIT
042064                          STARS
                                ;;*****************************************************************

                                        BGNSETUP        1

042064                                  BGNPTAB
042064  000000                          .WORD   0
042066  000004                          .WORD   L10151-./2-1
042070                          L10147:

042070  172440                          .WORD   CSRADR          ;DEFAULT CSR ADDRESS
042072  000224                          .WORD   VECADR          ;DEFAULT VECTOR ADDRESS
042074  000005                          .WORD   BRPRI           ;DEFAULT BR PRIORITY
042076  000000                          .WORD   0               ;DEFAULT DRIVE/CONTROLLER

042100                                  ENDPTAB
042100                          L10151:

042100                                  ENDSETUP

042100                                  ENDMOD

        000001                  .END
```

SYMBOL TABLE

```
ABOFLG  002234 G        BIT5   = 000040 G        CS2.PG= 002000 G        C$SVEC= 000037          E$END  = 002100
ADR    = 000020 G       BIT6   = 000100 G        CS2.UP= 020000 G        C$TPRI= 000013          E$LOAD= 000035
ASSEMB= 000010          BIT7   = 000200 G        CS2.U0= 000001 G        DIAGMC= 000000          FP.ACT  002736 G
ATOFLG  002230 G        BIT8   = 000400 G        CS2.U1= 000002 G        DLY     016042 G        FP.ADR  002732 G
AT3FLG  002232 G        BIT9   = 001000 G        CS2.U2= 000004 G        DLYCNT  003034 G        FP.BIT  002750 G
A.AS    002314 G        BLK    = 000040          CS2.WC= 040000 G        DMACNT  003036 G        FP.DPA  002744 G
A.BA    002302 G        BOE    = 000400 G        C$AU   = 000052         DMADLY  016016 G        FP.EXP  002734 G
A.BAE   002322 G        BRPRI  = 000005 G        C$AUTO= 000061          DPFLG   002242 G        FP.NAM  002730 G
A.CS1   002276 G        BWFLG   002244 G        C$BRK  = 000022         DRIFLG  003032 G        FP.NDX  002742 G
A.CS2   002306 G        BYTFLG  002236 G        C$BSEG= 000004          DRIVE   003000 G        FP.TBL  002746 G
A.CS3   002324 G        B.ABOR= 000004 G        C$BSUB= 000002          DSPBIT  017006 G        FP.XOR  002740 G
A.DT    002320 G        B.AS    006064 G        C$CEFG= 000045          DSPDAT  016374 G        FS.F0  = 000400 G
A.FS    002304 G        B.ATO  = 000001 G        C$CLCK= 000062          DSPDPA  016510 G        FS.F1  = 001000 G
A.IB    002312 G        B.AT3  = 000002 G        C$CLEA= 000012          DSPFIL  016536 G        FS.F2  = 002000 G
A.IS    002310 G        B.BIT   006134 G        C$CLOS= 000035          DSPREG  016054 G        FS.F3  = 004000 G
A.OB    002316 G        B.BYTE= 000010 G        C$CLP1= 000006          DSPSTA  016164 G        FS.F4  = 010000 G
A.SIZ  = 000030 G        B.CS1   005512 G        C$CVEC= 000036          D.AS    002264 G        FS.F5  = 020000 G
A.TBL   002276 G        B.CS2   005702 G        C$DCLN= 000044          D.BA    002252 G        FS.F6  = 040000 G
A.WC    002300 G        B.FS    005602 G        C$DODU= 000051          D.BAE   002272 G        FS.F7  = 100000 G
BA.AS   002454 G        B.IS    005776 G        C$DRPT= 000024          D.BDP   002516 G        FS.ST0= 000001 G
BA.BA   002442 G        B.REVA= 000200 G        C$DU   = 000053         D.BLOC  002506 G        FS.ST1= 000002 G
BA.BAE  002462 G        B.UPAR= 000020 G        C$EDIT= 000003          D.CS1   002246 G        FS.ST2= 000004 G
BA.CS1  002436 G        CBLFLG  003050 G        C$ERDF= 000055          D.CS2   002256 G        FS.ST3= 000010 G
BA.CS2  002446 G        CLRDRI  015464 G        C$ERHR= 000056          D.CS3   002274 G        FS.ST4= 000020 G
BA.CS3  002464 G        CLRISR  015324 G        C$ERRO= 000060          D.DT    002270 G        FS.ST5= 000040 G
BA.DT   002460 G        CLRMBC  015342 G        C$ERSF= 000054          D.FILP  002504 G        FS.ST6= 000100 G
BA.FS   002444 G        CLRTRA  015264 G        C$ERSO= 000057          D.FIL1  014371 G        FS.ST7= 000200 G
BA.IB   002452 G        CLRTST  015550 G        C$ESCA= 000010          D.FIL2  014437 G        F$AU   = 000015
BA.IS   002450 G        CLR.CM= 000011 G        C$ESEG= 000005          D.FS    002254 G        F$AUTO= 000020
BA.OB   002456 G        CPUFLG  003012 G        C$ESUB= 000003          D.IB    002262 G        F$BGN  = 000040
BA.SIZ= 000030 G        CPUID  = 177764 G        C$ETST= 000001          D.IS    002260 G        F$CLEA= 000007
BA.TBL  002436 G        CR     = 000015          C$EXIT= 000037          D.LPCN  002510 G        F$DU   = 000016
BA.WC   002440 G        CSRADR= 172440 G        C$GETB= 000026          D.MAX   002512 G        F$END  = 000041
BFR     005054 G        CSRFLG  003042 G        C$GETW= 000027          D.OB    002266 G        F$HARD= 000004
BFRSIZ= 001000 G        CS1.A1= 001000 G        C$GMAN= 000043          D.SIZ  = 000030 G        F$HW   = 000013
BITCNT  003006 G        CS1.DV= 004000 G        C$GPHR= 000042          D.TBL   002246 G        F$INIT= 000006
BIT0   = 000001 G        CS1.F0= 000002 G        C$GPLO= 000030          D.WC    002250 G        F$JMP  = 000050
BIT00  = 000001 G        CS1.F1= 000004 G        C$GPRI= 000040          D.WDP   002514 G        F$MOD  = 000000
BIT01  = 000002 G        CS1.F2= 000010 G        C$INIT= 000011          EF.CON= 000036 G        F$MSG  = 000011
BIT02  = 000004 G        CS1.F3= 000020 G        C$INLP= 000020          EF.NEW= 000035 G        F$PROT= 000021
BIT03  = 000010 G        CS1.F4= 000040 G        C$MANI= 000050          EF.PWR= 000034 G        F$PWR  = 000017
BIT04  = 000020 G        CS1.GO= 000001 G        C$MEM  = 000031         EF.RES= 000037 G        F$RPT  = 000012
BIT05  = 000040 G        CS1.IE= 000100 G        C$MSG  = 000023         EF.STA= 000040 G        F$SEG  = 000003
BIT06  = 000100 G        CS1.MC= 020000 G        C$OPEN= 000034          EOL     005350 G        F$SOFT= 000005
BIT07  = 000200 G        CS1.PS= 002000 G        C$PNTB= 000014          ERRBLK  005174 G        F$SRV  = 000010
BIT08  = 000400 G        CS1.RD= 000200 G        C$PNTF= 000017          ERRCNT  003004 G        F$SUB  = 000002
BIT09  = 001000 G        CS1.SC= 100000 G        C$PNTS= 000016          ERRFLG  003026 G        F$SW   = 000014
BIT1   = 000002 G        CS1.TR= 040000 G        C$PNTX= 000015          ERRMSG  005172 G        F$TEST= 000001
BIT10  = 002000 G        CS2.BA= 000010 G        C$OIO  = 000377         ERRNBR  005170 G        F.ADR1  012626 G
BIT11  = 004000 G        CS2.CL= 000040 G        C$RDBU= 000007          ERRNDX  003002 G        F.ADR2  012650 G
BIT12  = 010000 G        CS2.DL= 100000 G        C$REFG= 000047          ERRTYP  005166 G        F.BITS  014210 G
BIT13  = 020000 G        CS2.IR= 000100 G        C$RESE= 000033          ER.ADR  014452 G        F.BW    013544 G
BIT14  = 040000 G        CS2.MD= 000400 G        C$REVI= 000003          ER.DAT  015172 G        F.CBL   013464 G
BIT15  = 100000 G        CS2.MX= 001000 G        C$RFLA= 000021          ER.DMP  015074 G        F.CRLF  014214 G
BIT2   = 000004 G        CS2.NE= 010000 G        C$RPT  = 000025         ER.INI  014640 G        F.CSR   013660 G
BIT3   = 000010 G        CS2.OR= 000200 G        C$SEFG= 000046          ER.REG  015016 G        F.DAT1  013166 G
BIT4   = 000020 G        CS2.PA= 000020 G        C$SPRI= 000041          EVL    = 000004 G        F.DAT2  013244 G
```

SYMBOL TABLE

```
F.DMP1  012664 G        IN.OB   002376 G        L$DEPO  002011 G        L10014  020212          L10105  031170
F.DMP2  012716 G        IN.SIZ= 000030 G        L$DESC  005226 G        L10015  020226          L10106  031254
F.DMP3  012750 G        IN.TBL  002356 G        L$DES   002076 G        L10016  020232          L10107  031546
F.DMP4  013020 G        IN.WC   002360 G        L$DEVP  002060 G        L10017  020236          L10110  031434
F.DPAT  013624 G        ISR    = 000100 G        L$DISP  002124 G        L10020  020554          L10111  031534
F.ERCN  013304 G        IS.ATA= 100000 G        L$DLY   002116 G        L10021  020406          L10112  033542
F.MODE  013340 G        IS.AT0= 000001 G        L$DTP   002040 G        L10022  020546          L10113  032154
F.NAM   012607 G        IS.AT1= 000002 G        L$DTYP  002034 G        L10023  020700          L10114  032524
F.NED   013732 G        IS.AT2= 000004 G        L$DU    020230 G        L10024  021426          L10115  033114
F.PERR  013412 G        IS.AT3= 000010 G        L$DUT   002072 G        L10025  021006          L10116  033516
F.PWR   014000 G        IS.CYC= 000020 G        L$DVTY  005176 G        L10026  021054          L10117  034246
F.REG1  013056 G        IS.ERO= 002000 G        L$EF    002052 G        L10027  021232          L10120  034002
F.REG2  013132 G        IS.ERR= 040000 G        L$ENVI  002044 G        L10030  021420          L10121  034234
F.RPT1  014074 G        IS.ICP= 001000 G        L$ERRT  005166 G        L10031  022214          L10122  034766
F.RPT2  014164 G        IS.IDP= 000400 G        L$ETP   002102 G        L10032  021574          L10123  034514
F.UNIT  014036 G        IS.IP = 010000 G        L$EXP1  002046 G        L10033  021642          L10124  034754
F.VAL   012616 G        IS.IRY= 000200 G        L$EXP4  002064 G        L10034  022020          L10125  035334
G$CNTO= 000200          IS.XAB= 004000 G        L$EXP5  002066 G        L10035  022206          L10126  035170
G$DELM= 000372          IXE    = 004000 G        L$HARD  04   2 G        L10036  023004          L10127  035322
G$DISP= 000003          I$AU   = 000041          L$HIME  002120 G        L10037  022340          L10130  036604
G$EXCP= 000400          I$AUTO= 000041           L$HPCP  002016 G        L10040  022406          L10131  035634
G$HILI= 000002          I$CLN  = 000041          L$HPTP  002022 G        L10041  022572          L10132  036104
G$LOLI= 000001          I$DU   = 000041          L$HW    002214 G        L10042  022776          L10133  036346
G$NO   = 000000         I$HRD  = 000041          L$ICP   002104 G        L10043  023324          L10134  036572
G$OFFS= 000400          I$INIT= 000041           L$INIT  017322 G        L10044  023170          L10135  037012
G$OFSI= 000376          I$MOD  = 000041          L$LADP  002026 G        L10045  023306          L10136  037220
G$PRMA= 000001          I$MSG  = 000041          L$LAST  042064 G        L10046  023674          L10137  037506
G$PRMD= 000002          I$PROT= 000040           L$LOAD  002100 G        L10047  023504          L10140  040234
G$PRML= 000000          I$PTAB= 000041           L$LUN   002074 G        L10050  023666          L10141  040222
G$RADA= 000140          I$PWR  = 000041          L$MREV  002050 G        L10051  024050          L10142  041004
G$RADB= 000000          I$RPT  = 000041          L$NAME  002000 G        L10052  023776          L10143  040772
G$RADD= 000040          I$SEG  = 000041          L$PRIO  002042 G        L10053  024042          L10144  041126
G$RADL= 000120          I$SETU= 000041           L$PROT  017314 G        L10054  024516          L10145  041176
G$RADO= 000020          I$SFT  = 000041          L$PRT   002112 G        L10055  024164          L10146  041360
G$XFER= 000004          I$SRV  = 000041          L$REPP  002062 G        L10056  024240          L10147  042070
G$YES  = 000010         I$SUB  = 000041          L$REV   002010 G        L10057  024360          L10151  042100
HELP   = 000000         I$TST  = 000041          L$RPT   017172 G        L10060  024510          MAXTST= 000033 G
HLPFLG  003016 G        J$JMP  = 000167          L$SOFT  041300 G        L10061  025036          MBCFLG  003030 G
HOE    = 100000 G       LF     = 000012          L$SPC   002056 G        L10062  024650          M.ABT1  006375 G
HRDTBL  002772 G        LINBUF  005260 G         L$SPCP  002020 G        L10063  024740          M.ABT2  006447 G
IBE    = 010000 G       LINE    005256 G         L$SPTP  002024 G        L10064  025024          M.ABT3  006520 G
IBUF    003054 G        LOE    = 040000 G        L$STA   002030 G        L10065  025316          M.ABT4  006572 G
IDU    = 000040 G       LOGUNI  002774 G         L$SW    002226 G        L10066  025204          M.ADR1  006234 G
IER    = 020000 G       LOT    = 000010 G        L$TEST  002114 G        L10067  025304          M.ADR2  006263 G
INIFLG  003014 G        LP.BDP  040414           L$TIML  002014 G        L10070  027532          M.ASB1  006314 G
INTFLG  003022 G        LP.BXF  031606           L$UNIT  002012 G        L10071  025776          M.ATA1  006643 G
INTSRV  015226 G        LP.UAT  023054           L10000  002224          L10072  026412          M.ATA2  006706 G
IN.AS   002374 G        LP.WDP  037644           L10001  002246          L10073  027044          M.ATA3  006740 G
IN.BA   002362 G        LP.WXF  025364           L10002  014636          L10074  027506          M.ATA4  007005 G
IN.BAE  002402 G        L$ACP   002110 G         L10003  015014          L10075  030232          M.ATA5  007060 G
IN.CS1  002356 G        L$APT   002036 G         L10004  015072          L10076  027766          M.ATA6  007135 G
IN.CS2  002366 G        L$AU    020234 G         L10005  015170          L10077  030220          M.BYT1  007500 G
IN.CS3  002404 G        L$AUT   002070 G         L10006  015216          L10100  030746          M.BYT2  007533 G
IN.DT   002400 G        L$AUTO  020212 G         L10007  015224          L10101  030474          M.BYT3  007602 G
IN.FS   002364 G        L$CCP   002106 G         L10010  015232          L10102  030734          M.BYT4  007650 G
IN.IB   002372 G        L$CLEA  020214 G         L10011  017312          L10103  031266          M.CSR1  007210 G
IN.IS   002370 G        L$CO    002032 G         L10013  020210          L10104  031100          M.CSR2  007237 G
```

SYMBOL TABLE

```
M.DLT1  007707 G        N.BAE   005472 G        REG1    002754 G        TRAP4   015220 G        T10.2   024652
M.ERO1  007751 G        N.CS1   005352 G        REG2    002756 G        TRPFLG  003024 G        T10.3   024742
M.ERO2  010017 G        N.CS2   005412 G        REG3    002760 G        TSTINT  015444 G        T11     025040 G
M.ERR1  010064 G        N.CS3   005502 G        REG4    002762 G        TSTMBC  015410 G        T11.1   025106
M.ERR2  010116 G        N.DT    005462 G        REG5    002764 G        T$ARGC= 000001          T11.2   025206
M.FSR1  010163 G        N.FS    005402 G        REG6    002766 G        T$CODE= 007130          T12     025320 G
M.ICP1  010450 G        N.IB    005432 G        REG7    002770 G        T$ERRN= 000001          T12.1   025364
M.ICP2  010535 G        N.IS    005422 G        REVFLG  002226 G        T$EXCP= 000000          T12.2   026000
M.ICP3  010623 G        N.OB    005452 G        RIE.CM= 000171 G        T$FLAG= 000040          T12.3   026414
M.IDP1  010215 G        N.SIZ = 000140 G        RIE2.C= 000177 G        T$FREE= 042100          T12.4   027046
M.IDP2  010277 G        N.TBL   005352 G        RP.ERR  002620 G        T$GMAN= 000000          T13     027534 G
M.IDP3  010365 G        N.WC    005362 G        RP.PAS  002520 G        T$HILI= 000007          T13.1   027572
M.INT1  010706 G        OBUF    004054 G        RP.SIZ= 000100 G        T$LAST= 000001          T13.2   027770
M.INT2  010761 G        ONEFIL= 000001          R.AS    002344 G        T$LOLI= 000000          T14     030234 G
M.IRY1  011034 G        O$APTS= 000000          R.BA    002332 G        T$LSYM= 010000          T14.1   030272
M.IRY2  011102 G        O$AU  = 000000          R.BAE   002352 G        T$LTNO= 000033          T14.2   030476
M.IRY3  011162 G        O$BGNR= 000001          R.CS1   002326 G        T$NEST= 177777          T15     030750 G
M.IRY4  011246 G        O$BGNS= 000001          R.CS2   002336 G        T$NS0 = 000000          T15.1   031016
M.IRY5  011320 G        O$DU  = 000000          R.CS3   002354 G        T$NS1 = 000005          T15.2   031102
M.IRY6  011363 G        O$ERRT= 000000          R.DT    002350 G        T$NS2 = 000002          T15.3   031172
M.IRY7  011426 G        O$GNSW= 000001          R.ERR   002470 G        T$PCNT= 000000          T16     031270 G
M.IRY8  011500 G        O$POIN= 000001          R.FS    002334 G        T$PTAB= 010150          T16.1   031336
M.ISR1  011543 G        O$SETU= 000001          R.IB    002342 G        T$PTHV= 000001          T16.2   031436
M.ISR2  011572 G        PARFLG  002240 G        R.IS    002340 G        T$PTNU= 000001          T17     031550 G
M.MBC1  011650 G        PERFLG  003052 G        R.MAX   002466 G        T$SAVL= 177777          T17.1   031606
M.MBC2  011717 G        PNT   = 001000 G        R.OB    002346 G        T$SEGL= 177777          T17.2   032156
M.RDY1  011750 G        PQ.CSR  041176          R.SAD   002474 G        T$SIZE= 000006          T17.3   032526
M.RDY2  012024 G        PQ.DRI  041256          R.SDR   002502 G        T$SUBN= 000000          T17.4   033116
M.REG1  012076 G        PQ.PRI  041236          R.SIN   002476 G        T$TAGL= 177777          T18     033544 G
M.REG2  012134 G        PQ.VEC  041216          R.SIZ = 000030 G        T$TAGN= 010152          T18.1   033606
M.SW4   012153 G        PRI   = 002000 G        R.SRH   002500 G        T$TEMP= 000000          T18.2   034004
M.SW5   012233 G        PRI00 = 000000 G        R.SST   002472 G        T$TEST= 000033          T19     034250 G
M.SW6   012313 G        PRI01 = 000040 G        R.TBL   002326 G        T$TSTM= 177777          T19.1   034312
M.TRE1  012371 G        PRI02 = 000100 G        R.WC    002330 G        T$TSTS= 000001          T19.2   034516
M.UAT1  012454 G        PRI03 = 000140 G        SELECT  003010 G        T$$AU = 010017          T2      020556 G
M.UAT2  012531 G        PRI04 = 000200 G        SELMAS= 000033          T$$AUT= 010014          T20     034770 G
M.WRD1  007271 G        PRI05 = 000240 G        SELTES= 000010          T$$CLE= 010015          T20.1   035026
M.WRD2  007324 G        PRI06 = 000300 G        SETISR  015274 G        T$$DAT= 010151          T20.2   035172
M.WRD3  007373 G        PRI07 = 000340 G        SETRAP  015234 G        T$$DU = 010016          T21     035336 G
M.WRD4  007441 G        PS    = 177776 G        SILOCN  003040 G        T$$HAR= 010145          T21.1   035374
NA.AS   002424 G        PT.CSR  002720 G        SILODL  016030 G        T$$HW = 010000          T21.2   035636
NA.BA   002412 G        PT.DRI  002726 G        SQ.BW   041726          T$$INI= 010013          T21.3   036106
NA.BAE  002432 G        PT.PRI  002724 G        SQ.DP   041666          T$$MSG= 010006          T21.4   036350
NA.CS1  002406 G        PT.VEC  002722 G        SQ.REV  041360          T$$PC = 000001          T22     036606 G
NA.CS2  002416 G        PWRFLG  003046 G        SQ.SW4  041410          T$$PRO= 010012          T23     037014 G
NA.CS3  002434 G        Q.BBL   014314 G        SQ.SW5  041450          T$$PTA= 010150          T24     037222 G
NA.DT   002430 G        Q.BDP   014246 G        SQ.SW6  041510          T$$RPT= 010011          T25     037510 G
NA.FS   002414 G        Q.HLP1  014334 G        SQ.SW7  041562          T$$SOF= 010146          T25.1   037702
NA.IB   002422 G        Q.HLP2  014402 G        SQ.SW8  041622          T$$SRV= 010010          T26     040236 G
NA.IS   002420 G        Q.WBL   014274 G        SVCGBL= 000000          T$$SUB= 010143          T26.1   040452
NA.OB   002426 G        Q.WDP   014220 G        SVCINS= 000000          T$$SW = 010001          T27     041006 G
NA.SIZ= 000030 G        RDBLK   015642 G        SVCSUB= 000000          T$$TES= 010144          T3      020702 G
NA.TBL  002406 G        RDINI   015726 G        SVCTAG= 000000          T1      020240 G        T3.1    020742
NA.WC   002410 G        RDREG   015606 G        SVCTST= 000000          T1.1    020240          T3.2    021010
NEDFLG  003044 G        RD.CMD= 000071 G        S$LSYM= 010000          T1.2    020410          T3.3    021056
N.AS    005442 G        RD2.CM= 000077 G        TAB   = 000011          T10     024520 G        T3.4    021234
N.BA    005372 G        REG0    002752 G        TABFLG  003020 G        T10.1   024566          T4      021430 G
```

```
ZDRMA0 DR70 REPAIR DIAGNOSTIC   MACRO M1200  23-MAY-83 15:51  PAGE 216-4
SYMBOL TABLE

T4.1    021470          T5.4    022574          T8.1    023736          UAM    = 000200 G       WR.CMD= 000061 G
T4.2    021576          T6      023006 G        T8.2    024000          UNITPO  002776 G        WR2.CM= 000067 G
T4.3    021644          T6.1    023054          T9      024052 G        VECADR= 000224 G        X$ALWA= 000000
T4.4    022022          T6.2    023172          T9.1    024052          WIE.CM= 000161 G        X$FALS= 000040
T5      022216 G        T7      023326 G        T9.2    024166          WIE2.C= 000167 G        X$OFFS= 000400
T5.1    022274          T7.1    023366          T9.3    024242          WRTBLK  015674 G        X$TRUE= 000020
T5.2    022342          T7.2    023506          T9.4    024362          WRTINI  015762 G        $PATCH  041760 G
T5.3    022410          T8      023676 G

. ABS.  042100     000
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  28461 WORDS  ( 112 PAGES)
DYNAMIC MEMORY:  17282 WORDS  ( 66 PAGES)
ELAPSED TIME:  00:07:01
ZDRMA0.BIN/DS:GBL/EN:ABS:AMA,ZDRMA0/-SP/CR=SVC33/ML,ZDRMA0.P11
```