```
                        .TITLE CNOMBAO DMV11 MCTRL DIAG #2
                        .SBTTL PROGRAM DOCUMENT
                        .REM !


                                        I D E N T I F I C A T I O N

                        PRODUCT CODE:   AC 1828A MC

                        PRODUCT NAME:   CNOMBAO DMV11 MICRO-CONTROLLER STATIC DIAGNOSTIC PART 2

                        PRODUCT DATE:   APRIL 1984

                        MAINTAINER:     ISS DIAGNOSTICS

                        AUTHORS:        CHRIS BRIENEN
                                        RAY MARSHALL

                        MODIFIED BY:    JAKI BERG       9 APR 1984

                        PURPOSE:        THIS DIAGNOSTIC IS DESIGNED TO PERFORM STATIC LOGIC TESTS FOR
                                        THE M8053 OR M8064 (HEREAFTER REFERRED TO AS THE DMV OR DMV-11)



                                THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
                                NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
                                EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO
                                RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

                                NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF
                                SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS
                                AFFILIATED COMPANIES.

                                COPYRIGHT (C) 1984 BY DIGITAL EQUIPMENT CORPORATION

                        THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

                                DIGITAL         PDP             UNIBUS          MASSBUS
                                DEC             DECUS           DECTAPE
```

```
49
50
51                                  ***** MODIFICATION HISTORY *****
52
53      REV A: ORIGINAL RELEASE                  CHRIS BRIENEN, RAY MARSHALL     14-JAN-81
54
55      REV B: INSTALLED OUTSTANDING PATCHES                                     11-JUL-83
56
57      CVDMBB -> CNDMBA                                    JAKI BERG            9-APR-84
58      CHANGES WERE MADE TO  CVDMBB  TO PRODUCE  CNDMBA  FOR THE  FALCON-PLUS PROJECT
59      (SBC-11/21+).   CHANGES, MARKED BY ";JB REV A-0", ARE:
60        - SET THE  ODT BREAK  VECTOR (LOCATION 140) TO THE STARTING ADDRESS OF
61          FALCON'S ODT ROM (170000-OCTAL).
62          FALCON + CANNOT BE STRAPPED TO MODE 0 ON POWER UP.  THE "STRAPPED TO MODE 0"
63          QUESTION WAS REMOVED.  THE DEFAULTS AND P-TABLE WERE SET APPROPRIATELY.
64
```

PROGRAM DOCUMENT

CONTENTS

1.0 INTRODUCTION

THE M8053 AND M8064 ARE SINGLE-LINE SYNCHRONOUS, MICRO-PROCESSOR BASED COMMUNICATIONS INTERFACES WHICH CAN SUPPORT BOTH CHARACTER-ORIENTED (DDCMP, BSC, ETC.) AND BIT-ORIENTED (SDLC, HDLC, ETC.) PROTOCOLS. THE PURPOSE OF THIS PROGRAM IS TO PERFORM DIAGNOSTIC TESTING OF THE CSRS, RAM, AND BASIC MICRO-PROCESSOR LOGIC ON THESE BOARDS. THE FOLLOWING FUNCTIONS WILL BE PERFORMED: DMV RESIDENT U-DIAG EXECUTION CSR ADDRESSING, VIA REGISTER STATIC BIT INTERACTION AND READ/WRITE TESTING, AND ON-BOARD RAM TESTING.

THE STATIC LOGIC TESTS WILL PROVIDE EXTENSIVE TROUBLESHOOTING CAPABILITIES, SUCH AS TIGHT SCOPE LOOPS, SWITCH OPTIONS, AND ABILITY TO "LOCK" ONTO INTERMITTENT ERRORS. IN ADDITION TESTS ARE DESIGNED AND STRUCTURED TO ACHIEVE MAXIMUM FAULT RESOLUTION AND FACILITATE REPLACEMENT OF THE SMALLEST FIELD REPLACEABLE UNIT.

THIS PROGRAM IS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR AND A STRUCTURED PROGRAMMING APPROACH. BECAUSE THE DESIGN CONFORMS TO THE SUPERVISOR (STANDALONE VERSION) THE PROGRAM IS COMPATIBLE WITH ACT, APT, XXDP+, AND SLIDE.

THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM ALLOWS MODIFICATION OF DEVICE PARAMETERS, SUCH AS LSI-BUS ADDRESS, VECTOR ADDRESSES AND DEVICE PRIORITY. IN ADDITION, THE OPERATOR CAN SPECIFY PARTICULAR TESTS TO BE RUN AND A VARIETY OF LOOPING, RUNNING, AND REPORTING MODES.

DEVICE ERRORS WILL BE REPORTED AS THEY OCCUR. THE REPORT WILL INCLUDE A TEST NUMBER AND DESCRIPTION OF THE ERROR, GOOD AND BAD TEST DATA, AND APPLICABLE DEVICE REGISTER CONTENTS.

2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE M8053/8064 STATIC LOGIC TESTS:

SBC-11/21+
16K WORDS OF MEMORY
CONSOLE TERMINAL
M8053 OR M8064 COMMUNICATIONS INTERFACE

3.0 PRELIMINARY PROGRAM REQUIREMENTS

THIS PROGRAM (CNDMB) SHOULD BE THE SECOND OF THE FIVE DMV 11 STATIC DIAGNOSTICS TO BE RUN (CNDMA SHOULD BE RUN FIRST). ERRORS FOUND IN THIS PROGRAM SHOULD BE CORRECTED BEFORE RUNNING ANY OF THE LINE UNIT DIAGNOSTICS (CNDMC, CNDMD, OR CNDME).

4.0 GENERAL PROGRAM CONSIDERATIONS

4.1 DIAGNOSTIC SUPERVISOR

THIS PROGRAM IS COMPATIBLE WITH THE STANDALONE DIAGNOSTIC SUPERVISOR, AND MUST BE LOADED TO BE CO-RESIDENT WITH THE SUPERVISOR, OR BE PREVIOUSLY COMBINED WITH THE SUPERVISOR AND LOADED AS A SINGLE FILE. IN EITHER CASE, THE COMBINED PROGRAM WILL NOT EXCEED 16K OF MEMORY.

4.2 EXECUTION TIME

THE MAXIMUM TIME REQUIRED TO RUN THIS PROGRAM IS ABOUT ONE MINUTE PER PASS FOR EACH UNIT.

4.3 XXDP+

THIS PROGRAM MAY BE LOADED UNDER XXDP+, AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.4 ACT/SLIDE

THIS PROGRAM MAY BE LOADED UNDER ACT OR SLIDE AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.5 APT

THIS PROGRAM MAY BE LOADED BY THE APT SYSTEM (INCLUDING APT-RD) AND RUN IN PROGRAM MODE OR SCRIPT MODE.

4.6 MEMORY MANAGEMENT

MEMORY MANAGEMENT IS UTILIZED IN THIS PROGRAM TO VERIFY THE DMV 11'S ABILITY TO NPR INTO (AND OUT OF) EXTENDED MEMORY.

4.7 ERROR LOGGING

AT THE END OF EACH PASS ON ALL UNITS, THE PROGRAM PRINTS OUT THE CUMULATIVE TOTAL NUMBER OF ERRORS SINCE THE LAST START OR RESTART COMMAND.

5.0 PROGRAM LOAD MEDIA

THIS PROGRAM CAN BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER OR FROM ACT, SLIDE, OR APT SYSTEMS, OR FROM ANY MEDIA SUPPORTED BY XXDP+. WHEN USING THE PAPER TAPE ABSOLUTE LOADER, THE PROGRAM SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC SUPERVISOR. WHEN USING XXDP+, THE

DIAGNOSTIC SUPERVISOR SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC PROGRAM.

6.0 OPERATING INSTRUCTIONS

6.1 LOADING AND STARTING PROCEDURES

6.1.1 LOADING PROCEDURES

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM ANY XXDP+ LOAD MEDIA. WHEN LOADED UNDER XXDP+, THE DIAGNOSTIC SUPERVISOR WILL BE LOADED AUTOMATICALLY.

6.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC PROCEDURES TO START THE PROGRAM.

6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

THE DIAGNOSTIC CAN BE EXECUTED STANDALONE UNDER XXDP+, WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:

A) LOAD AND START DIAGNOSTIC USING RUN COMMAND
B) RECEIVE DIAGNOSTIC SUPERVISOR IDENTIFICATION AND PROMPT (DRS-C>)
C) ENTER STA<CR>
D) ANSWER HARDWARE AND SOFTWARE QUESTIONS
E) GET END OF PASS MESSAGES OR ERROR MESSAGES
F) TO END EXECUTION, ENTER CONTROL/C

6.2 INITIAL DIALOGUE

AFTER THE PROGRAM AND THE SUPERVISOR ARE LOADED AND THE PROGRAM IS STARTED, THE FOLLOWING IDENTIFICATION IS TYPED :

```
DRS LOADED
DIAG. RUN-TIME SERVICES
CNDMB-A-0
DMV-11 U-CONTRL LOGIC DIAG - PART 2 OF 2
UNIT IS M8053 OR M8064
DR>
```

THE OPERATOR THEN PROCEEDS BY TYPING ONE OR MORE OF THE COMMANDS DESCRIBED IN THE FOLLOWING SECTION 6.3. (FOR MORE DETAILED INFORMATION, REFER TO THE DIAGNOSTIC SUPERVISOR FUNCTIONAL SPECIFICATION).

6.3 PROGRAM OPTIONS

6.3.1 START COMMAND

```
*****************************************************************
STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
   <FLAG-LIST>/EOP:<INCR>
*****************************************************************
```

6.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)

<TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1;2 ETC.) OR RANGES OF DECIMAL NUMBERS (1 5;8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS. ON THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY THE OPERATOR. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING EXECUTION. IN THIS CASE EXIT FROM THE PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR BY OCCURANCE OF AN ERROR WITH THE HALT ON ERROR FLAG BEING SET. THE EXIT IS A RETURN TO COMMAND MODE. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG LIST>)

<FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG-1>, OR <FLAG-0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

HOE	HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED
LOE	LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR
IER	INHIBIT ERROR REPORTING
IBE	INHIBIT BASIC ERROR REPORTS
IXE	INHIBIT EXTENDED ERROR REPORTS
PRI	DIRECT ALL MESSAGES TO A LINE PRINTER
PNT	PRINT NUMBER OF TEST BEING EXECUTED
BOE	BELL ON ERROR
UAM	RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
ISR	INHIBIT STATISTICAL REPORTS
IDU	INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

LOT LOOP ON TEST

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

<INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.5 EFFECT OF START COMMAND

THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND THEN THE DIAGNOSTIC TESTS THEMSELVES.

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION "0 UNITS?" TO WHICH THE OPERATOR REPLIES WITH A DECIMAL NUMBER N FROM 1 TO 16. THE TERM "UNIT" REFERS TO THE DEVICE TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL BE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE UNIT. THE OPERATOR MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION. HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE AFTER THE PARENTHESES.

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.

WHEN THE QUESTION "0 UNITS?" IS ANSWERED, MEMORY STORAGE IS ALLOCATED FOR THE P-TABLES, AND IF THERE IS NOT ENOUGH TO ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED. IN THIS CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO TEST ALL UNITS.

EXAMPLE:

```
STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:MOE=1:UAM:LOE
```

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, EACH PASS CONSISTING OF TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST ALL UNITS. THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON A COMMAND OTHER THAN START TO CLEAR A FLAG THAT WAS

PREVIOUSLY SET. NOTE THAT ON ALL COMMANDS ONLY THE FIRST THREE LETTERS ARE SCANNED.

6.3.2 RESTART COMMAND

```
**************************************************************
RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
   <FLAG-LIST>/UNITS:<UNIT-LIST>
**************************************************************
```

6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

6.3.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (0,1 ETC.) OR RANGES OF DECIMAL NUMBERS (0-5, 8-10 ETC.) THAT SPECIFY THE UNITS TO BE TESTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS MAY RANGE FROM 0 THRU N-1 (N IS THE NUMBER OF UNITS SPECIFIED IN THE PREVIOUS START COMMAND). THE NUMBER INDICATES THE POSITION OF THE P-TABLE AS THE DATA WAS ENTERED DURING THE HARDWARE DIAGLOGUE. THE UNITS WHICH ARE SELECTED MUST NOT HAVE BEEN DROPPED BY THE DROP COMMAND. SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW. DEFAULT IS TO TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP COMMAND.

6.3.2.3 EFFECT OF RESTART COMMAND

THE RESTART COMMAND DIFFERS FROM THE START COMMAND IN THAT THE P-TABLES FROM THE PREVIOUS START COMMAND (THERE MUST HAVE BEEN ONE) ARE USED, INSTEAD OF NEW ONES BEING BUILT. THE UNITS SWITCH GIVES THE ABILITY TO SELECT A SUBSET OF THESE. THE SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED (OPERATOR WILL BE ASKED). THE COMMAND CAN BE USED AFTER COMMAND MODE HAS BEEN REENTERED IN ANY OF THE THREE NORMAL WAYS: A) THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B) AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

6.3.3 CONTINUE COMMAND

```
**************************************************************
CON(TINUE)/PASS:<PASS-CNT/FLAGS:<FLAG-LIST>
**************************************************************
```

6.3.3.1 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS

THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART.
IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

6.3.3.2 FLAG SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS SAME AS IN START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

6.3.3.3 EFFECT OF CONTINUE COMMAND

CONTINUE MUST FOLLOW A START OR RESTART, AND COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

6.3.4 PROCEED COMMAND

****
PRO(CEED)/FLAGS:<FLAG-LIST>
****

6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

6.3.4.2 EFFECT OF PROCEED COMMAND

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE. COMMAND MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE PARAMETERS MAY BE ALTERED.

6.3.5 ADD COMMAND

****
ADD/UNITS:<UNIT-LIST>
****

6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.5.2 EFFECT OF ADD COMMAND

THE UNITS SPECIFIED ARE ADDED TO THE TEST SEQUENCE. EACH

UNIT MUST HAVE A P-TABLE IN MEMORY DUE TO AN EARLIER HARDWARE DIALOGUE. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR CONTINUE. THE UNITS SWITCH MUST BE SPECIFIED. THE ADD COMMAND IS MEANINGFUL ONLY FOR UNITS THAT WERE PREVIOUSLY DROPPED.

6.3.6 DROP COMMAND

****
DRO(P)/UNITS:<UNIT-LIST>
****

6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.6.2 EFFECT OF DROP COMMAND

THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING. THE UNITS WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START COMMAND. THE UNITS SWITCH MUST BE ENTERED. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.

6.3.7 PRINT COMMAND

****
PRI(NT)
****

6.3.7.1 EFFECT OF PRINT COMMAND

THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

6.3.8 DISPLAY COMMAND

****
DIS(PLAY)/UNITS:<UNIT-LIST>
****

6.3.8.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.8.2 EFFECT OF DISPLAY COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS

THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.

6.3.9 FLAGS COMMAND

****
FLA(GS)
****

6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

6.3.10 ZFLAGS COMMAND

****
ZFL(AGS)
****

6.3.10.1 EFFECT OF ZFLAGS COMMAND

ALL FLAGS ARE CLEARED.

6.3.11 CONTROL CHARACTERS

A CONTROL C (C) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO COMMAND MODE.

A CONTROL Z (Z) ENTERED DURING ONE OF THE THREE OPERATOR DIALOGUES- HARD CORE QUESTIONS (SEE 6.2),HARDWARE DIALOGUE (SEE 6.3.1.5), OR SOFTWARE DIALOGUE (SEE 6.3.1.5) CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

A CONTROL O (O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES ALL TELETYPE OUTPUT TO BE SURPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

6.3.12 HARDWARE PARAMETERS

THE FOLLOWING 3 QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONSE.

1. DEVICE CSR ADDRESS : (O) 160020?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SEL0) RESIDE ON THE LSI-BUS. THE ALLOWABLE RANGE IS 160020-177760 (OCTAL), AND THE DEFAULT VALUE IS 160020.

2. DEVICE VECTOR ADDRESS : (0) 300 ?

THIS IS THE ADDRESS OF THE INPUT INTERRUPT VECTOR FOR THIS DEVICE. THE ALLOWABLE RANGE IS 000-674 (OCTAL), AND THE DEFAULT VALUE IS 300.

3. DEVICE PRIORITY LEVEL : (0) 4 ?

THIS IS THE CPU PRIORITY AT WHICH THE INTERRUPT HANDLERS OF THIS DEVICE WILL BE EXECUTED. THE ALLOWABLE RANGE IS 0-7, AND THE DEFAULT VALUE IS 4.

4. BOARD TYPE (0=M8064, 1=M8053-V35, 2=M8053-EIA) : (0) 0 ?

THIS IS THE TYPE OF DMV-11 CURRENTLY INSTALLED. NOTE THAT THE M8053 IS SWITCH SELECTABLE BETWEEN V.35 AND EIA.

5. IS THIS A MANUFACTURING TEST STAND : (L) N ?

THIS QUESTION REFERS TO A SPECIFIC MEMORY CONFIGURATION THAT IS REQUIRED TO RUN TEST #8 (SEE SEC. 7.0).

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY THIS PROGRAM.

6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SOON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES. ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

ON THE FIRST TRIP THRU THE QUESTIONS, ALL OF THE SLOTS IN ALL OF THE P-TABLES ARE FILLED. IF THE OPERATOR TYPES IN LESS THAN N EXPLICIT VALUES IN RESPONSE TO A PARTICULAR QUESTION, THESE VALUES ARE PLACED IN THE P-TABLES (ONE VALUE GOING INTO THE PROPER SLOT OF EACH P-TABLE BEGINNING WITH THE FIRST P-TABLE) UNTIL THE STRING OF VALUES IS EXHAUSTED. THE LAST VALUE IN THE STRING BECOMES THE NEW DEFAULT AND IS USED TO FILL THAT SLOT IN THE REMAINING P-TABLES.

ON SUBSEQUENT TRIPS THRU THE QUESTIONS, THE SAME PROCESS IS CARRIED OUT, EXCEPT THAT THE EARLIEST P-TABLE NOT TO HAVE RECEIVED AN EXPLICIT VALUE IN ANY OF ITS SLOTS NOW ASSUMES THE ROLE THAT TABLE NUMBER ONE PLAYED IN THE FIRST TRIP.

THE SERIES OF QUESTIONS IS REISSUED UNTIL AT LEAST ONE QUESTION HAS RECEIVED N EXPLICIT VALUES FROM THE OPERATOR.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST

```
684                     NAMED VALUE.
685
686                     A  STRING  OF  VALUES  MAY  BE  GIVEN  AS  A RANGE (6-10 FOR
687                     EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA,  THIS
688                     SAMPLE   RANGE   TRANSLATES   TO   THE   STRING  6,7,8,9,10  (AN
689                     INCREMENT OF 1). IF THE VALUES  ARE  ADDRESSES,  THE  SAMPLE
690                     RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).
691
692                     NOW  LET  US  SEE  HOW  WE  COULD  USE THESE CAPABILITIES TO
693                     CONSTRUCT A SET OF P-TABLES.  ASSUME THAT WE HAVE 16  UNITS,
694                     AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE
695                     SLOTS  IN  THE  P-TABLE,  THREE  HARDWARE  QUESTIONS  IN  THE
696                     DIALOGUE).  LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE
697                     THE NUMBER 75 FOR ALL 16 TABLES.  LET THE DESIRED VALUE  FOR
698                     THE    SECOND    PARAMETER    BE    EQUAL   TO   THE   UNIT   NUMBER
699                     (0,1,2,...,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE  THE
700                     VALUE  11.  LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE
701                     THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE NUMBER  77  FOR
702                     THE LAST 9 UNITS.
703
704                     THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:
705
706                     # UNITS (D) ? 16
707                     UNIT 0
708                     <QUESTION 1> ? 75
709                     <QUESTION 2> ? 0-6
710                     <QUESTION 3> ? 76
711
712                     UNIT 7
713                     <QUESTION 1> ?
714                     <QUESTION 2> ? 7-11,,13-15
715                     <QUESTION 3> ? 77
716
717                     THE   FIRST   TIME  THE  SERIES  IS  ASKED,  SLOT  ONE  RECEIVES  A  75
718                     IN  ALL  16  TABLES.   SLOT  TWO  RECEIVES  THE  VALUES  0,1,2,...,6
719                     IN   TABLES  0  THRU  6  AND  A  CONSTANT  6  IN  TABLES  7  THRU  15.
720                     SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.
721
722                     THE  SECOND  TIME  THRU  THE  SERIES,  TABLES  7  THRU  THE  END   ARE
723                     GOING  TO  BE  AFFECTED  (NOTE  THAT  THIS  PIECE  OF  INFORMATION  IS
724                     PRINTED  OUT  FOR  THE  THE  OPERATOR  IN  THE  FORM   "UNIT   XX"   AT
725                     THE   BEGINNING   OF   EACH   SERIES).   QUESTION  1  IS  RESPONDED  TO
726                     BY  A  <CR>,  SO  SLOT  ONE  STAYS  AT  CONSTANT   75   IN   TABLES   7
727                     THRU  15,  SINCE  NO  NEW  EXPLICIT  VALUES  ARE  TYPED  IN.   SLOT  TWO
728                     GETS  THE  VALUES  7,8,9,10,11  IN  TABLES  7   THRU   11,   AND
729                     GETS  AN  11  IN  SLOT  12,  AND  GETS  THE  VALUES  13,14,15  IN
730                     TABLES  13  THRU  15.  SLOT  THREE  GETS  THE  VALUE  77  IN  TABLES  7
731                     THRU 15.
732
733                     THE  DIALOGUE  IS  TERMINATED  WHEN  THE  SOFTWARE  RECOGNIZES  THAT
734                     16  EXPLICIT  VALUES  HAVE  BEEN  GIVEN  FOR  AT  LEAST  ONE  QUESTION
735                     (NAMELY QUESTION 2).
736
737
```

```
                        7.0 TEST DESCRIPTIONS


;*************************************************************************************
;*      TEST 1  <VIA TIMER 2 ONE SHOT MODE>
;*
;*    THIS  TEST  VERIFIES  THAT  THE  TIMER 2 COUNTER IS OPERATIONAL IN
;*    INTERVAL-TIMER (ONE-SHOT) MODE.
;*
;*      THE FOLLOWING IS PERFORMED :
;*
;*    A  MASTER CLEAR IS DONE & THE TIMER IS PLACED IN INTERVAL-TIMER MODE
;*    BY  SETTING  ACR5 = 0 AND THE PROGRAM CHECKS FOR "T2" (BIT 5 IN IFR)
;*    TO BE INITIALLY CLEARED.
;*
;*    T2L-L  (ADR  08)  & T2C-H (ADR 09) ARE BOTH LOADED WITH 252 (OCTAL).
;*    (THIS  IS  EQUIVALENT  TO  AAAA  (HEX) OR 43,690 (DECIMAL).) LOADING
;*    T2C-H STARTS THE COUNTER.
;*
;*    T2L-L  IS  LOADED  WITH 001 AND T2C-H IS LOADED WITH 000 IN ORDER TO
;*    SET  "T2"  WITH  A QUICK UNDERFLOW. THE "T2" FLAG BIT IN IFR IS READ
;*    AND CHECKED TO BE SET.
;*
;*    T2C-H IS CHECKED TO = 0. CHECKING T2C H SHOULD NOT HAVE CLEARED "T2"
;*    -- THIS IS VERIFIED.
;*
;*    T2C-L  IS CHECKED TO = 0. CHECKING T2C-L SHOULD HAVE CLEARED "T2" --
;*    THIS TOO IS VERIFIED.
;*
;*    T2C-H  IS  LOADED  WITH  0 AGAIN TO INITIATE A NEW COUNT DOWN (WHICH
;*    SHOULD  UNDERFLOW  ALMOST  IMMEDIATELY)  AND  THE "T2" BIT IN IFR IS
;*    CHECKED TO BE SET AGAIN.
;*
;*    T2L-L IS LOADED WITH 125 (OCTAL) AND "T2" BIT IS CHECKED TO BE STILL
;*    SET.
;*
;*    T2C-H IS LOADED WITH 125, AND THE "T2" BIT IS READ AND CHECKED TO BE
;*    CLEARED BY THE LOADING OF T2C-H.
;*
;*
;*************************************************************************************


;*************************************************************************************
;*      TEST 2  <VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE>
;*
;* A MASTER CLEAR IS DONE. THEN THE SHIFT REG IS PLACED IN INPUT MODE
;* UNDER CONTROL OF VIA CLK, BY SETTING ACR BIT 4 TO 0, BIT 3 TO 1, AND BIT 2
;* TO 0. THE PROGRAM CHECKS FOR THE SR FLAG (BIT 2) IN THE IFR TO BE INITIALLY
;* CLEARED. THEN, THE SR IS LOADED TO INITIALIZE THE SR OPERATION, AND THE
;* PROGRAM CHECKS FOR SR FLAG = 1 AFTER ABOUT 8 US. AND READS SR REGISTER TO
;* VERIFY THAT SHIFTING OCCURRED.
;*************************************************************************************
```

```
                        ;*****************************************************************************
                        ;*      TEST 3  <NPR CONTROL REGISTER - MASTER CLEAR>
                        ;*
                        ;* THE PROGRAM SETS THE FOLLOWING BITS IN THE NPR CONTROL REGISTER :
                        ;* IN/OUT, BYTE OPER, AND DISABL INIT. THE REGISTER IS READ AND VERIFIED.
                        ;* THEN, A MASTER CLEAR IS PERFORMED, AND THE REGISTER IS READ AND CHECKED FOR
                        ;* 000.
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 4  <NPR DATA-OUT>
                        ;*
                        ;* FIRST SUBTEST :
                        ;* THE NPR OUTPUT ADDRESS REGISTEF IS LOADED WITH THE ADDRESS OF A 2 BYTE
                        ;* BUFFER IN THE PROGRAM. THEN, EACH WORD OF DATA PATTERN F IS LOADED INTO THE
                        ;* NPR OUTPUT DATA REGISTER, A FULLWORD NPR OUTPUT REQUEST IS PERFORMED.
                        ;* AND THE PROGRAM CHECKS FOR THE CORRECT DATA IN THE PROGRAM BUFFER. ALSO,
                        ;* THE PROGRAM CHECKS THAT THE ABORT XFER BIT IN THE NPR CONTROL REGISTER
                        ;* NEVER GETS SET.
                        ;*   DATA PATTERN F = 125252, 052525, 000000, 177777, 000001, 000002, 000004,
                        ;*                    000010, 000020, 000040, 000100, 000200, 000400, 001000,
                        ;*                    002000, 004000, 010000, 020000, 040000, 100000, 177776,
                        ;*                    177775, 177773, 177767, 177757, 177737, 177677, 177577,
                        ;*                    177377, 176777, 175777, 173777, 167777, 157777, 137777,
                        ;*                    077777, 000000
                        ;*
                        ;* SECOND SUBTEST:
                        ;* THE ABOVE OPERATIONS ARE REPEATED IN BYTE NPR TRANSFER MODE, USING THE DATA
                        ;* BYTES IN DATA PATTERN B. THE LOW BYTE OF THE PROGRAM BUFFER IS USED, AND
                        ;* THE UPPER BYTE IS CLEARED AT THE START, AND IS CHECKED TO REMAIN UNCHANGED
                        ;* THROUGHOUT THE SUBTEST.
                        ;*   DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
                        ;*                    200, 376, 375, 373, 367, 357, 337, 277, 177, 000
                        ;*
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 5  <NPR DATA-IN>
                        ;*
                        ;* THE NPR INPUT ADDRESS REGISTER IS LOADED WITH THE ADDRESS OF A 2 BYTE
                        ;* BUFFER IN THE PROGRAM. THEN, EACH WORD OF DATA PATTERN F IS LOADED INTO THE
                        ;* PROGRAM BUFFER, A FULLWORD NPR INPUT REQUEST IS ISSUED AND PERFORMED.
                        ;* AND THE PROGRAM CHECKS FOR THE CORRECT DATA IN THE NPR INPUT DATA REG.
                        ;* ALSO, THE PROGRAM CHECKS THAT THE ABORT XFER BIT IN THE NPR CONTROL
                        ;* REGISTER NEVER GETS SET.
                        ;*   DATA PATTERN F = 125252, 052525, 000000, 177777, 000001, 000002, 000004,
                        ;*                    000010, 000020, 000040, 000100, 000200, 000400, 001000,
                        ;*                    002000, 004000, 010000, 020000, 040000, 100000, 177776,
                        ;*                    177775, 177773, 177767, 177757, 177737, 177677, 177577,
                        ;*                    177377, 176777, 175777, 173777, 167777, 157777, 137777,
                        ;*                    077777, 000000
                        ;*****************************************************************************
```

```
853                     ;*****************************************************************************
854                     ;*      TEST 6  <NPR XFER ABORT>
855                     ;*
856                     ;* FIRST SUBTEST :
857                     ;* THE PROGRAM PERFORMS AN OUTPUT NPR REQUEST TO A NON-EXISTENT MEMORY
858                     ;* LOCATION, AND CHECKS FOR THE ASSERTION OF ABORT XFER BIT IN THE NPR CONTROL
859                     ;* REGISTER. THEN, AN OUTPUT NPR IS DONE AND CHECKED, TO A LOCATION IN THE
860                     ;* PROGRAM, USING 125252 FOR DATA, AND THE PROGRAM CHECKS FOR ABORT XFER TO
861                     ;* BE CLEARED BY SETTING THE DONE BIT.
862                     ;* SECOND SUBTEST :
863                     ;* THE ABOVE SUBTEST IS REPEATED USING INPUT NPR'S.
864                     ;*****************************************************************************
865
866
867                     ;*****************************************************************************
868                     ;*      TEST 7  <NPR EXTENDED ADDRESS BIT TEST>
869                     ;*
870                     ;* THIS TEST WILL ONLY BE RUN IF THERE IS AT LEAST 32K WORDS OF MEMORY ON THE
871                     ;* SYSTEM. IF THERE IS, THE PROGRAM CHOOSES A LOCATION TO USE IN THE ADDRESS
872                     ;* RANGE 200000-377776 (OCTAL). THEN, THE FOLLOWING 2 SUBTESTS ARE PERFORMED :
873                     ;*
874                     ;* FIRST SUBTEST :
875                     ;* AN INPUT NPR IS PERFORMED AND CHECKED USING THE MEMORY LOCATION, WITH
876                     ;* 125252 FOR DATA. THE PROGRAM CHECKS THAT THE ABORT XFER BIT REMAINS
877                     ;* CLEARED.
878                     ;* SECOND SUBTEST :
879                     ;* AN OUTPUT NPR IS PERFORMED AND CHECKED USING THE MEMORY LOCATION, WITH
880                     ;* 125252 FOR DATA. THE PROGRAM CHECKS THAT THE ABORT XFER BIT REMAINS
881                     ;* CLEARED.
882                     ;*****************************************************************************
883
884
885                     ;*****************************************************************************
886                     ;*      TEST 8  <SPECIAL MFG EXTENDED BIT TEST>
887                     ;*
888                     ;* THIS TEST WAS DESIGNED SPECIFICALLY TO ALLOW MANUFACTURING TO CHECK THE
889                     ;* NPRAIX/NPRAOX BITS WITHOUT A FULL 4 M. OF MEMORY.
890                     ;*
891                     ;* IT WILL CHECK THE 12 DMV EXTENDED ADDRESS BITS (6:NPRAIX/6:NPRAOX) ON
892                     ;* A Q22 SYSTEM IF MEMORY IS PRESENT AT THE FOLLOWING PHYSICAL ADDRESSES:
893                     ;*
894                     ;*      17600000        17400000        17200000
895                     ;*      16600000        15600000        13600000
896                     ;*       7600000
897                     ;*
898                     ;* FIRST SUBTEST :      TEST "NPRAIX" EXTENDED ADDRESS BITS
899                     ;* SECOND SUBTEST :     TEST "NPRAOX" EXTENDED ADDRESS BITS
900                     ;*****************************************************************************
901
902
903                     ;*****************************************************************************
904                     ;*      TEST 9  <Q-BUS INTERRUPT "A" & "B" SELECTION>
905                     ;*
906                     ;* THIS  TEST  CONTAINS SUBTESTS IN WHICH A SEQUENCE OF STEPS IS
907                     ;* PERFORMED. IN GENERAL, EACH SUBTEST PERFORMS THE FOLLOWING:
908                     ;*
909                     ;*      1. INTERRUPTS ARE DISABLED FOR BOTH "A" & "B"
```

```
                        ;*
                        ;*      2. THE INTERRUPT REQUEST REGISTER IS WRITTEN INTO
                        ;*
                        ;*      3. A TEST IS MADE TO BE SURE THAT NEITHER INTERRUPT OCCURS
                        ;*
                        ;*      4. BOTH INTERRUPTS ARE ENABLES
                        ;*
                        ;*      5. A TEST IS MADE TO BE SURE THAT IF AN INTERRUPT IS EXPECTED, IT IS
                        ;*         RECEIVED AND IF IT ISN'T EXPECTED IT DOESN'T HAPPEN.
                        ;*
                        ;* ALL TESTING IS DONE HERE WITH THE PROCESSOR'S PRIORITY SET AT 0.
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 10 <BUS RESET WITH DISABLE INIT SET> .PAGE
                        ;*
                        ;* A  BYTE SELECT REGISTER (BSEL3) IS LOADED WITH 377, DISABLE INIT BIT IS SET
                        ;* IN  THE  NPR CONTROL REGISTER, AND A BUS RESET INSTRUCTION IS EXECUTED. THE
                        ;* PROGRAM  THEN  CHECKS  THAT THE DMV-11 WAS NOT CLEARED, BY CHECKING FOR 377
                        ;* STILL IN BSEL3
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 11 <MASTER CLEAR WITH DISABLE INIT SET>
                        ;*
                        ;* THE "DISABL INIT" BIT IN THE NPR CONTROL REGISTER IS SET AND A MASTER CLEAR
                        ;* IS  ISSUED.  IF  THE  MASTER  CLEAR SUBROUTINE DETECTS AN ERROR, THE MASTER
                        ;* CLEAR  WILL  NOT  HAVE  FUNCTIONED PROPERLY, WHERE THE NORMAL ERROR MESSAGE
                        ;* (QUEUED  UP  BY "MASCLR") IS NORMALLY PRINTED, THIS TEST WILL PRINT ITS OWN
                        ;* INSTEAD.
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 12 <DCOK H LO BIT>
                        ;*
                        ;* DCOK H LO IS SET IN THE NPR CONTROL REGISTER WHICH SHOULD CAUSE A VECTOR TO
                        ;* THE  FIRST  INTERRUPT  HANDLER  WHERE THE VECTOR IS CHANGED TO POINT TO THE
                        ;* SECOND HANDLER. THIS SECOND HANDLER WILL THEN STALL FOR A WHILE WAITING FOR
                        ;* THE  POWER-UP  INTERRUPT  WHICH  SHOULD KICK US INTO THE SECOND HANDLER. IN
                        ;* BOTH  HANDLERS  FLAGS  ARE  SET  TO  SAY THAT WE GOT THERE. WHEN WE FINALLY
                        ;* RETURN  TO  OUR MAINLINE CODE, WE WILL RESUME THE DELAY FUNCTION WE WERE IN
                        ;* AND THEN CHECK THE FLAGS.
                        ;*
                        ;* IN SUBTEST # 1, WE EXPECT THE DMV TO BE RESET.
                        ;*
                        ;*****************************************************************************


                        ;*****************************************************************************
                        ;*      TEST 13 <HALT MODE VERIFICATION>
                        ;*
                        ;* THIS TEST CONTAINS TWO (2) SUBTESTS DESIGNED TO VERIFY THE FUNCTIONALITY
                        ;* OF  THE  "HALT"  CONTROL CONTAINED WITHIN THE NPR CONTROL REGISTER. IN EACH
                        ;* CASE,  MICROCODE  IS  LOADED INTO THE DMV IN ORDER TO CONTROL THE TESTING
```

```
                        ;* FROM THERE.
                        ;*
                        ;*-------------------- ----------------------------------------------------------
                        ;*
                        ;* SUBTEST # 1:
                        ;*
                        ;* HERE WE VERIFY THAT WE CAN CONTROL NPR'S AND DCOK PROPERLY WHILE THE 11 CPU
                        ;* IS HALTED.
                        ;*
                        ;*      11 CPU'S OPERATIONS:                    DMV-11'S OPERATIONS:
                        ;*
                        ;* THE MICROCODE IS MOVED INTO THE DMV.
                        ;*
                        ;* CLEAR TMP0.  THIS WILL BE OUR TEST
                        ;* LOCATION FOR THE NPR OPERATION.
                        ;*
                        ;* SETUP FOR POWER-FAIL VECTORING THROUGH
                        ;* LOCATION 24.
                        ;*
                        ;* THE MICROCODE IS INITIATED & BSEL7 IS
                        ;* SET TO -1 AS A FLAG.
                        ;*
                        ;* WAIT FOR BSEL7 TO BE CLEARED            CLEAR BSEL7 AND WAIT FOR IT TO GO
                        ;*                                         NON-ZERO AGAIN.  THIS PUTS THE
                        ;*                                         DMV IN SYNC. WITH THE 11 CPU
                        ;*
                        ;* SAVE R6 IN OLDSP FOR RECOVERY LATER.
                        ;* CLEAR TMP0, LOAD INTO SEL4 THE
                        ;* ADDRESS OF TMP0, AND SET BSEL7 TO -1.
                        ;*
                        ;* START LOOPING -- INCREMENTING TMP0      GET THE ADDRESS OF TMP0 FROM SEL6
                        ;*                                         AND SAVE IT FOR LATER
                        ;*
                        ;*                                         HALT THE 11 CPU.
                        ;*
                        ;* CONSOLE "ODT" SHOULD BE ENTERED.        NPR-IN THE CURRENT CONTENTS OF TMP0
                        ;*                                         & PUT IT INTO SEL4 (THE FULL WORD).
                        ;*
                        ;*                                         DELAY FOR ABOUT 100 MICROSECONDS
                        ;*                                         (THE TIME ISN'T CRITICAL).
                        ;*
                        ;* THE 11 CPU SHOULD NOT BE EXECUTING
                        ;* ANYTHING NOW -- NOT EVEN "ODT"
                        ;*
                        ;*                                         DROP THE "HALT" SIGNAL TO RELEASE
                        ;*                                         THE 11 CPU AND SET "DCOK H LO" &
                        ;*                                         "DISABL INIT", DROP "DCOK H LO"
                        ;*
                        ;*
                        ;* WE SHOULD GO
                        ;* THROUGH A POWER-UP SEQUENCE.   R6 IS
                        ;* RESTORED FROM OLDSP, INTERRUPT
                        ;* PRIORITY LEVEL IS RESTORED TO 0, &
                        ;* INTERRUPT VECTOR 24 IS RETURNED TO
                        ;* THE DIAGNOSTIC SUPERVISOR.  SEL4 IS
                        ;* COMPARED AGAINST TMP0 -- THEY SHOULD
                        ;* BE EQUAL.
```

```
1024                    ;*
1025                    ;* NOW CLEAR BSEL7.                    WAIT FOR BSEL7 TO CLEAR. THEN DROP
1026                    ;*                                     "DISABLE INIT" AND EXIT TO M-LOOP.
1027                    ;*
1028                    ;*-------------------------------------------------------------------------
1029                    ;*
1030                    ;* SUBTEST # 2:
1031                    ;*
1032                    ;* HERE  THE  HALT MODE IS USED IN A WAY WHICH VERY CLOSELY MATCHES THE DMV-11
1033                    ;* MICROCODE'S  UTILIZATION  DURING  A  "MOP  BOOT"  OPERATION. THE INTERRUPT
1034                    ;* VECTOR  AREA  IS COMPLETELY OVERWRITTEN BY THE DMV NPR'S AND IS THEREFORE
1035                    ;* BACKED  UP  ELSEWHERE  IN  THE  11 CPU'S MEMORY. THERE IS ALSO THE POSSIBLE
1036                    ;* CONTENTION  WITH  THE  DIAGNOSTIC  SUPERVISOR  --  TO HELP HERE, AS MUCH AS
1037                    ;* POSSIBLE WILL BE DONE AT INTERRUPT LEVEL 7.
1038                    ;*
1039                    ;*      11 CPU'S OPERATIONS:                        DMV-11'S OPERATIONS:
1040                    ;*
1041                    ;* THE MICROCODE IS MOVED INTO THE DMV.
1042                    ;*
1043                    ;* THE INTERRUPT VECTOR AREA IS BACKED-
1044                    ;* UP IN AN I/O BUFFER FOLLOWING THE
1045                    ;* PROGRAM
1046                    ;*
1047                    ;* THE MICROCODE IS INITIATED & BSEL7 IS
1048                    ;* SET TO -1 AS A FLAG.
1049                    ;*
1050                    ;* WAIT FOR BSEL7 TO BE CLEARED        CLEAR BSEL7 AND WAIT FOR IT TO GO
1051                    ;*                                     NON-ZERO AGAIN PUTTING BOTH
1052                    ;*                                     PROCESSORS IN SYNC. WITH EACH
1053                    ;*                                     OTHER
1054                    ;* CLEAR TMP0 AND SAVE R6 FOR RECOVERY
1055                    ;* LATER.  SET BSEL7 AGAIN AND WAIT FOR
1056                    ;* TMP0 TO BE SET.
1057                    ;*                                     SET HALT, "DCOK H LO", & "DISABL
1058                    ;*                                     INIT" AND PERFORM 2 NOP'S AS A
1059                    ;*                                     1 MICROSECOND DELAY
1060                    ;*
1061                    ;* ENTRY INTO THE CONSOLE "ODT" WILL   CLEAR "DCOK H LO", SET "HALT" &
1062                    ;* BE INITIATED.                       "DISABL INIT"
1063                    ;*
1064                    ;*                                     NPR-OUT THE FOLLOWING:
1065                    ;*                                      LOC:  CONTENTS
1066                    ;*                                       24   000000  VECTOR TO LOC 0
1067                    ;*                                       26   000340  @ PRIORITY 7
1068                    ;*                                        0   012700  MOV  #-1,R0
1069                    ;*                                        2   177777
1070                    ;*                                        4   000777  BR   .
1071                    ;*                                     ----------------------------
1072                    ;*
1073                    ;* THE ENTRY INTO THE CONSOLE "ODT"    SET "DCOK H LO" & "DISABL INIT"
1074                    ;* WILL BE ABORTED!                    AND CLEAR "HALT"
1075                    ;*
1076                    ;*                                     DELAY FOR A SHORT TIME (ABOUT 1
1077                    ;*                                     MICROSECOND
1078                    ;*
1079                    ;* THE ROUTINE NPR'D INTO LOC. 0 WILL  CLEAR "DCOK H LO" AND SET "DISABL
1080                    ;* BE EXECUTED -- EVENTUALLY HANGING   INIT"
```

```
1081                    ;* AT THE "BR  ." INSTRUCTION @ LOC. 4.
1082                    ;*
1083                    ;*                                      NPR-OUT THE FOLLOWING:
1084                    ;*                                       LOC:  CONTENTS
1085                    ;*                                         6   005001  CLR  R1
1086                    ;*                                        10   062701  ADD  #(.+2),R1
1087                    ;*                                        12   062701  ADD  #(.+2),R1
1088                    ;*                                         .     .      .
1089                    ;*                                         .     .      .
1090                    ;*                                         .     .      .
1091                    ;*                                       360   062701  ADD  #(.+2),R1
1092                    ;*                                       362   062701  ADD  #(.+2),R1
1093                    ;*                                       364   010037  MOV  R0,@#TMP0
1094                    ;*                                       366   [TMP0]
1095                    ;*                                       370   013706  MOV  @#OLDSP,R6
1096                    ;*                                       372   [OLDSP]
1097                    ;*                                       374   000137  JMP  HLTST2
1098                    ;*                                       376   [HLTST2]
1099                    ;*                                      ----------------------------
1100                    ;*                                      THIS IS SYNONYMOUS TO THE DMV-11
1101                    ;*                                      LOADING A "MESSAGE" STARTING AT MEM.
1102                    ;*                                      LOC. 000006.
1103                    ;*
1104                    ;*                                      NPR-OUT THE FOLLOWING:
1105                    ;*                                       LOC:  CONTENTS
1106                    ;*                                         4   000240  NOP
1107                    ;*                                      ----------------------------
1108                    ;*                                      THIS IS HOW THE DMV-11 WILL TAKE THE
1109                    ;*                                      11 CPU OUT OF THE "BR  ." CONDITION.
1110                    ;* THE ROUTINE JUST LOADED BY THE DMV
1111                    ;* MICROCODE WILL NOW BE EXECUTED (WE    DROP "DISABL INIT"
1112                    ;* HOPE).  WHEN THE SUBROUTINE IS RE-
1113                    ;* ENTERED (@ HLTST2),                   AN EXIT IS TAKEN TO THE M-LOOP
1114                    ;*
1115                    ;*   1  THE INTERRUPT VECTORS WILL BE
1116                    ;*      RESTORED;
1117                    ;*   2  THE PRIORITY LEVEL WILL BE
1118                    ;*      LOWERED BACK TO 0;
1119                    ;*   3  R0, R1, & R6 WILL ALL BE CHECKED
1120                    ;*      FOR THE PROPER CONTENTS; AND
1121                    ;*   4  TMP0 WILL BE CHECKED FOR THE
1122                    ;*      PROPER CONTENTS;
1123                    ;*************************************************************************
1124
1125
1126
```

8.0 ERROR INFORMATION

8.1 ERROR REPORTING

ERRORS ARE REPORTED BY THE PROGRAM AS THEY OCCUR (IF NOT INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR ERROR REPORT FORMAT, AND CONSISTS OF A DESCRIPTION OF THE ERROR, THE TEST NUMBER, SUBTEST NUMBER, PC OF THE ERROR CALL, DEVICE ADDRESS, AND BASIC AND EXTENDED ERROR INFORMATION.

THE FOLLOWING EXAMPLE PROVIDES A TYPICAL ERROR REPORT, WHICH DESCRIBES A "MASTER CLEAR FAILURE" ERROR, AND PROVIDES THE PC OF THE ERROR CALL AND THE DEVICE REGISTER CONTENTS :

```
CNDMB DVC FTL ERR 00001 ON UNIT 00 TST 002 SUB 000 PC: 021122
MASTER CLEAR FAILURE

  THE CONTENTS OF ALL BYTE SELECT REG'S ARE:
BSEL0   BSEL1   BSEL2   BSEL3
 000     000     000     000
    BSEL4   BSEL5   BSEL6   BSEL7
     000     000     121     000
BSEL10  BSEL11  BSEL12  BSEL13
 000     000     000     000
    BSEL14  BSEL15  BSEL16  BSEL17
     000     000     000     000
```

FOR OTHER ERRORS, THE REPORT MAY BE MORE EXTENSIVE, AND REQUIRE ADDITIONAL DATA TO BE REPORTED.

IF EXTENDED ERROR INFORMATION HAD BEEN INHIBITED USING THE IXE FLAG PRIOR TO RUNNING THE TEST, THE ABOVE ERROR WOULD HAVE BEEN REPORTED IN THE FOLLOWING SHORTENED FORM :

```
CNDMB DVC FTL ERR 00001 ON UNIT 00 TST 002 SUB 000 PC: 021122
MASTER CLEAR FAILURE
```

K2

LISTING & ASSEMBLY CONTROL

```
                        .SBTTL LISTING & ASSEMBLY CONTROL

000000                  HELP=0          ; CONTROL LISTING OF HELP INFORMATION
                                        ; HELP=0   NO LIST
                                        ; HELP=1   LIST

002000                          .=2000

                                .MCALL  SVC
002000                          SVC                             ; INITIALIZE SUPERVISOR MACROS

002000                          BGNMOD  LU1MOD


000001                  $LSTIN= 1
000001                  $LSTTAG= 1
000001                  SVCINS= 1       ; LIST INSTRUCTIONS, SHIFTED RIGHT
000001                  SVCTST= 1       ; LIST TEST TAGS, SHIFTED RIGHT
000001                  SVCSUB= 1       ; LIST SUBTEST TAGS, SHIFTED RIGHT
000001                  SVCGBL= 1       ; LIST GLOBAL TAGS, SHIFTED RIGHT
000001                  SVCTAG= 1       ; LIST OTHER TAGS, SHIFTED RIGHT

                        ;       CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
                        ;       TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS.  CHANGE THE
                        ;       SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS.  YOU MAY
                        ;       CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.

002000                  POINTER BGNAU,BGNDU,ERRTBL
```

L2

PROGRAM HEADER

```
                        .SBTTL PROGRAM HEADER
                        ;++
                        ;THE PROGRAM HEADER MACRO CHARACTERIZES THIS DIAGNOSTIC. THE
                        ;HEADER MACRO'S ARGUMENTS ARE FILE NAME, RELEASE LEVEL, PATCH
                        ;DISPOSITION OF THE MOST RECENT PATCH, MAXIMUM TEST TIME IN SEC.,
                        ;AND THE TYPE OF DIAGNOSTIC (0-SEQUENTIAL, 1-EXERCISER).  THESE
                        ;ARGUMENTS ARE IN RESPECTIVE ORDER.
                        ;--


                        HEADER CNDMB,A,0,60.,0
002000                                                  L$NAME::
002000  103                                                     .ASCII /C/
002001  116                                                     .ASCII /N/
002002  104                                                     .ASCII /D/
002003  115                                                     .ASCII /M/
002004  102                                                     .ASCII /B/
002005  000                                                     .BYTE   0
002006  000                                                     .BYTE   0
002007  000                                                     .BYTE   0
002010                                                  L$REV::
002010  101                                                     .ASCII  /A/
002011                                                  L$DEPO::
002011      060                                                 .ASCII  /0/
002012                                                  L$UNIT::
002012  000000                                                  .WORD   0
002014                                                  L$TIML::
002014  000074                                                  .WORD   60.
002016                                                  L$HPCP::
002016  036344                                                  .WORD   L$HARD
002020                                                  L$SPCP::
002020  000000                                                  .WORD   0
002022                                                  L$HPTP::
002022  002160                                                  .WORD   L$HW
002024                                                  L$SPTP::
002024  000000                                                  .WORD   0
002026                                                  L$LADP::
002026  037054                                                  .WORD   L$LAST
002030                                                  L$STA::
002030  000000                                                  .WORD   0
002032                                                  L$CO::
002032  000000                                                  .WORD   0
002034                                                  L$DTYP::
002034  000000                                                  .WORD   0
002036                                                  L$APT::
002036  000000                                                  .WORD   0
002040                                                  L$DTP::
002040  002124                                                  .WORD   L$DISPATCH
002042                                                  L$PRIO::
002042  000000                                                  .WORD   0
002044                                                  L$ENVI::
002044  000000                                                  .WORD   0
002046                                                  L$EXP1::
002046  000000                                                  .WORD   0
002050                                                  L$MREV::
002050      003                                                 .BYTE   C$REVISION
002051      003                                                 .BYTE   C$EDIT
```

M2

PROGRAM HEADER

```
                002052
002052  000000  002052
002054  000000  002054
                002056
002056  000000  002056
                002060
002060  003254  002060
                002062
002062  000000  002062
                002064
002064  000000  002064
                002066
002066  000000  002066
                002070
002070  023120  002070
                002072
002072  023114  002072
                002074
002074  000000  002074
                002076
002076  003274  002076
                002100
002100  104035  002100
                002102
002102  002202  002102
                002104
002104  022020  002104
                002106
002106  023076  002106
                002110
002110  022752  002110
                002112
002112  022012  002112
                002114
002114  000000  002114
                002116
002116  000000  002116
                002120
002120  000000  002120
1223
1229                            .EVEN
```

SEQ 0025

```
L$EF::
                .WORD   0
                .WORD   0
L$SPC::
                .WORD   0
L$DEVP::
                .WORD   L$DVTYP
L$REPP::
                .WORD   0
L$EXP4::
                .WORD   0
L$EXP5::
                .WORD   0
L$AUT::
                .WORD   L$AU
L$DUT::
                .WORD   L$DU
L$LUN::
                .WORD   0
L$DESP::
                .WORD   L$DESC
L$LOAD::
                EMT     E$LOAD
L$ETP::
                .WORD   L$ERRTB
L$ICP::
                .WORD   L$INIT
L$CCP::
                .WORD   L$CLEAN
L$ACP::
                .WORD   L$AUTO
L$PRT::
                .WORD   L$PROT
L$TEST::
                .WORD   0
L$DLY::
                .WORD   0
L$HIME::
                .WORD   0
```

N2

DISPATCH TABLE

```
1231                                    .SBTTL DISPATCH TABLE
1232
1233                                    ;////////////////////////////////////////////////////////////////////////////
1234                                    ;/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
1235                                    ;/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
1236                                    ;////////////////////////////////////////////////////////////////////////////
1237
1238 002122                                    DISPATCH        13.
     002122  000015                                                         .WORD   13
     002124                                                         L$DISPATCH::
     002124  023122                                                 .WORD   T1
     002126  024132                                                 .WORD   T2
     002130  025020                                                 .WORD   T3
     002132  025406                                                 .WORD   T4
     002134  026116                                                 .WORD   T5
     002136  026366                                                 .WORD   T6
     002140  026774                                                 .WORD   T7
     002142  031[illegible]34                                       .WORD   T8
     002144  0325[illegible]4                                       .WORD   T9
     002146  03404[illegible]                                       .WORD   T10
     002150  03413[illegible]                                       .WORD   T11
     002152  03420[illegible]                                       .WORD   T12
     002154  03475[illegible]                                       .WORD   T13
1239
```

```
                        .SBTTL DEFAULT HARDWARE P-TABLE

                        ;//////////////////////////////////////////////////////////////////////
                        ;/ THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                        ;/ THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                        ;/ IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
                        ;//////////////////////////////////////////////////////////////////////

002156                          BGNHW   DFPTBL
002156  000010                                                          .WORD   L10000-L$HW/2
002160                                                                  L$HW::
002160                                                                  DFPTBL::

002160  160020                  .WORD   160020          ;DMV11 CSR UNIBUS ADDRESS
002162  000300                  .WORD      300          ;DMV11 INTERRUPT VECTOR
002164  004000                  .WORD     4000          ;DMV11 INTERRUPT PRIORITY LEVEL = 4
002166  000000                  .WORD      000          ;SWITCH REG. #1 (BOOT ADDRESS)
002170  000000                  .WORD      000          ;SWITCH REG. #2 (DDCMP ADDRESS)
002172  000000                  .WORD        0          ;MODULE IS M8064
002174  000000                  .WORD        0          ;H3254&H3255 USED
002176  000011                  .WORD   000011          ;MISC. CONTROLS:

                        ; POWER-UP MODE 0 MASK = 100
                        ;   0 = NOT JUMPERED FOR MODE 0 POWER-UP  <--- DEFAULT SETTING  ;JB REV A-0
                        ;   1 = JUMPERED FOR MODE 0 POWER-UP
                        ;       BOTH W5 & W6 REMOVED

                        ; MFG EXTENDED MEMORY CONFIGURATION MASK = 200
                        ;   0 = NORMAL TESTING
                        ;   1 = Q22 SYSTEM WITH MEMORY @ FOLLOWING LOCATIONS:
                        ;               17600000        17400000        17200000
                        ;               16600000        15600000        13600000
                        ;                7600000

002200                          ENDHW
002200                                                                  L10000:
```

```
1280                            .SBTTL SOFTWARE P-TABLE
1281
1282                            ;//////////////////////////////////////////////////////////////////////
1283                            ;/ THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
1284                            ;/ PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
1285                            ;//////////////////////////////////////////////////////////////////////
1286
1287 002200                             BGNSW   SFPTBL
     002200  000000                                                        .WORD   L10001-L$SW/2
     002202                                                         L$SW::
     002202                                                         SFPTBL::
1288 002202                             ENDSW
     002202                                                         L10001:
```

GLOBAL EQUATES SECTION

```
1290                            .SBTTL GLOBAL EQUATES SECTION
1291
1292
1293                            ;//////////////////////////////////////////////////////////////////////////////
1294                            ;/      THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
1295                            ;/      ARE USED IN MORE THAN ONE TEST.
1296                            ;//////////////////////////////////////////////////////////////////////////////
1297
1298 002202                           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
                                ;
                                ;                                   ; BIT POSITION IN SECOND STATUS WORD
                    000040      EF.START**      32.                 ; (100000) START COMMAND WAS ISSUED
                    000037      EF.RESTART**    31.                 ; (040000) RESTART COMMAND WAS ISSUED
                    000036      EF.CONTINUE**   30.                 ; (020000) CONTINUE COMMAND WAS ISSUED
                    000035      EF.NEW**        29.                 ; (010000) A NEW PASS HAS BEEN STARTED
                    000034      EF.PWR**        28.                 ; (004000) 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          PRIO3== 140
                000100          PRIO2== 100
                000040          PRIO1== 40
                000000          PRIO0== 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
1299
1300                            .SBTTL  DEFINE THE NUMBER OF CSR'S
1301            000010          CSREGS  =  8.
1302
1303                            ;*****************************************************************
1304
1305                            .SBTTL  NPR ADDRESS REGISTER EQUATES
1306            000070          NPRAOL  = 70                    ;OUT NPR ADRS LO REG
1307            000071          NPRAOH  = NPRAOL+1              ;OUT NPR ADRS HI REG
1308            000072          NPRAOX  = NPRAOL+2              ;OUT NPR EXTENDED ADRS REG
1309            000074          NPRAIL  = NPRAOL+4              ;IN NPR ADRS LO REG
1310            000075          NPRAIH  = NPRAOL+5              ;IN NPR ADRS HI REG
1311            000076          NPRAIX  = NPRAOL+6              ;IN NPR EXTENDED ADRS REG
1312            000200          NPRBS7  = BIT7                  ;"BANK SELECT 7" BIT -- W/IN EXTENDED ADRS. REG.
1313
1314
1315
1316                            .SBTTL  NPR DATA REG EQUATES
1317            123000          NPRDRL  = 123000                ;NPR DATA REGISTER -- LOW BYTE
1318            123001          NPRDRH  = NPRDRL+1              ;NPR DATA REGISTER -- HIGH BYTE
1319
1320
1321
1322                            .SBTTL  NPR CONTROL REG EQUATES
1323            123004          NPRCTL  = NPRDRL+4              ;NPR CONTROL REGISTER
1324            000200          NPRABT  = BIT7                  ;=1 IF BUS TIME-OUT ON NPR
1325            000100          NPRGO   = BIT6                  ;SET FOR NOP, CLEAR TO "GO" / 0=DONE, 1=BUSY
1326            000040          NPRIO   = BIT5                  ;0 = (LSI ==> DMV);  1 = (DMV ==> LSI)
1327            000020          LSIHLT  = BIT4                  ;SETTING THIS WILL "HALT" THE LSI-11 !!
1328            000010          NPRBYT  = BIT3                  ;SET TO 1 TO WRITE BYTE ONLY TO LSI-11
1329            000004          DMVPU   = BIT2                  ;SET BY MICRO-DIAG.  MUST REMAIN SET!!!
1330            000002          LSIDCL  = BIT1                  ;IF SET, WILL CAUSE POWER DOWN CONDITION IN LSI!
1331            000001          DMVDAI  = BIT0                  ;"DISABLE INIT" FROM EFFECTING DMV-11
1332
1333
1334                            .SBTTL  NPR REQUEST FUNCTIONS
```

NPR REQUEST FUNCTIONS

```
1335    000004          NPRLD   = DMVPU                 ;WORD XFER:  LSI ==> DMV
1336    000044          NPRDL   = DMVPU!NPRIO           ;WORD XFER:  DMV ==> LSI
1337    000054          NPRDLB  = DMVPU!NPRIO!NPRBYT    ;BYTE XFER:  DMV ==> LSI
1338
1339                    ;-------------------------------------------------------------------------------
1340
1341                    .SBTTL  INTERRUPT REG EQUATES
1342    123005          IRQREG  = 123005                ;INTERRUPT REQUEST REG
1343    000004          IRQA    = BIT2                  ;REQUEST BIT FOR XX0 INTERRUPT -- "A"
1344    000002          IRQB    = BIT1                  ;REQUEST BIT FOR XX4 INTERRUPT -- "B"
1345
1346                    ;-------------------------------------------------------------------------------
1347
1348                    .SBTTL  CONTROL FLAGS FROM P-TABLE ENTRIES
1349    000001          PU24    = BIT0                  ;POWER-FAIL VECTORING MODE.  1 = MODE 0
1350                                                    ;  (I.E. JUMPERS W5 & W6 BOTH REMOVED)
```

SWITCH PACKS

```
                        .SBTTL  SWITCH PACKS

                        ;;*****************************************************************************
                        ;* SWITCH PACKS
                        ;;*****************************************************************************

121000                  SWPBOT  = 121C00                ;"BOOT ADDRESS" SWITCH PACK [A200]
121400                  SWPDDCMP = 121400               ;"DDCMP ADDRESS" SWITCH PACK [A300]
```

CSR REG. DEFINITION FOR MAINT. LOOP

```
                        .SBTTL  CSR REG. DEFINITION FOR MAINT. LOOP

                        ;+*********************************************************************
                        .SBTTL          MAINTENANCE REGISTER - BSEL0
                        ;-*********************************************************************
                        ;       INTERRUPT ENABLE BITS

000001                  IENBA   = BIT0                  ;INTERRUPT ENABLE "A"
000020                  IENBB   = BIT4                  ;INTERRUPT ENABLE "B"


                        ;+*********************************************************************
                        .SBTTL          MAINTENANCE REGISTER - BSEL1
                        ;-*********************************************************************
                        ; MAINT. LOOP CONTROL BITS:

000200                  RUN     = BIT7
000100                  MCLR    = BIT6
000001                  MREQ    = BIT0


                        ;+*********************************************************************
                        .SBTTL          MAINTENANCE REGISTER - BSEL2
                        ;-*********************************************************************
                        ; MAINTENANCE FUNCTION CODES

000001                  REDLOC  = 1                     ;FUNCTION CODE FOR READ A 6502 LOCATION
000002                  WRILOC  = 2                     ;FUNCTION CODE FOR WRITE A 6502 LOCATION
000003                  REDPAG  = 3                     ;FUNCTION CODE FOR READ A 6502 MEMORY PAGE
000004                  WRIPAG  = 4                     ;FUNCTION CODE FOR WRITE A 6502 RAM PAGE
000005                  EXECUT  = 5                     ;FUNCTION CODE FOR EXECUTE AT GIVEN PC

000200                  MRDY    = BIT7                  ;M-LOOP REDY FOR A COMMAND WHEN SET
```

```
                                .SBTTL  DMV INTERNAL ADDRESSES

                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;               DMV INTERNAL ADDRESSES
                                ;---++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                ;00000000  <<  MICROPROCESSOR REGISTER ADDRESS EQUATES  >> 00000000

                                .SBTTL          BYTE & WORD SELECT REGISTERS

000020                          SLT0    =020
000020                          BSLT0   =SLT0
000021                          BSLT1   =SLT0+1
000022                          SLT2    =SLT0+2
000022                          BSLT2   =SLT0+2
000023                          BSLT3   =SLT0+3
000024                          SLT4    =SLT0+4
000024                          BSLT4   =SLT0+4
000025                          BSLT5   =SLT0+5
000026                          SLT6    =SLT0+6
000026                          BSLT6   =SLT0+6
000027                          BSLT7   =SLT0+7

                                .SBTTL          VIA'S REGISTERS

120000                          ORB     =120000
120001                          ORA     =ORB+1
120002                          DDRB    =ORB+2
120003                          DDRA    =ORB+3
120004                          T1CL    =ORB+4
120005                          T1CH    =ORB+5
120005                          T1LHGO  =ORB+5
120006                          T1LL    =ORB+6
120007                          T1LH    =ORB+7
120010                          T2LL    =ORB+10
120010                          T2CL    =T2LL
120011                          T2CH    =ORB+11
120012                          SR      =ORB+12
120013                          ACR     =ORB+13
120014                          PCR     =ORB+14
120015                          IFR     =ORB+15
120016                          IENR    =ORB+16
120017                          ORAM    =ORB+17

                                .SBTTL          VIA'S "IFR" REGISTER'S BIT ASSIGNMENTS

000200                          IFRIRQ  =BIT7   ;"IRQ" HAS BEEN ISSUED -- LOGICAL "OR" OF BITS 0 --> 6
000100                          IFRT1   =BIT6   ;"T1" -- TIMER # 1 TIMED-OUT
000040                          IFRT2   =BIT5   ;"T2" -- TIMER # 1 TIMED-OUT
000020                          IFRCB1  =BIT4   ;"CB1" EDGE DETECTED ("K2 LINE UNIT STEP" O/P SIGNAL FROM SR)
000010                          IFRCB2  =BIT3   ;"CB2" EDGE DETECTED (UNUSED!)
000004                          IFRSR   =BIT2   ;"SR" REGISTER COMPLETED SHIFT OPERATION
000002                          IFRCA1  =BIT1   ;"CA1" EDGE DETECTED ("K6 MOD RDY H")
000001                          IFRCA2  =BIT0   ;"CA2" EDGE DETECTED ("K2 CTS H")
```

J3

NTST -- BASIC TEXT FOR STARTING EACH TEST

```
1701                                    ;-----------------------------------------------------
```

GLOBAL DATA SECTION

```
                                .SBTTL GLOBAL DATA SECTION

                                ;////////////////////////////////////////////////////////////////////////////////
                                ;/      THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ;/      IN MORE THAN ONE TEST.
                                ;////////////////////////////////////////////////////////////////////////////////

                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                .SBTTL  CONTROL BLOCK FOR STACKED ERROR MESSAGES
                                ;---+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                        ERRTBL
002202                                                                  L$ERRTBL::
002202  000000                ERRTYP::        .WORD   0
002204  000000                ERRNBR::        .WORD   0
002206  000000                ERRMSG::        .WORD   0
002210  000000                ERRBLK::        .WORD   0

                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                .SBTTL  STORAGE FOR DEVICE REGISTERS
                                ;---+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
002212                          WSR0:
002212  000000                BSR0:   .WORD   0
002214                          WSR2:
002214  000000                BSR1:   .WORD   0
002216                          WSR4:
002216  000000                BSR2:   .WORD   0
002220                          WSR6:
002220  000000                BSR3:   .WORD   0
002222                          WSR10:
002222  000000                BSR4:   .WORD   0
002224                          WSR12:
002224  000000                BSR5:   .WORD   0
002226                          WSR14:
002226  000000                BSR6:   .WORD   0
002230                          WSR16:
002230  000000                BSR7:   .WORD   0
002232  000000                BSR10:  .WORD   0
002234  000000                BSR11:  .WORD   0
002236  000000                BSR12:  .WORD   0
002240  000000                BSR13:  .WORD   0
002242  000000                BSR14:  .WORD   0
002244  000000                BSR15:  .WORD   0
002246  000000                BSR16:  .WORD   0
002250  000000                BSR17:  .WORD   0

                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                .SBTTL  MISCELLANEOUS STORAGE
                                ;---+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
002252  000000                TDATA:  .WORD   0               ;TEST DATA
002254  000000                GDATA:  .WORD   0               ;EXPECTED DATA
002256  000000                BDATA:  .WORD   0               ;ACTUAL DATA
002260  000000                XDATA:  .WORD   0               ;EXCLUSIVE OR BETWEEN "GDATA" & "BDATA"
002262  110400                DELAY1: .WORD   110400          ;DELAY TIME, 3 INST., 500 MILLISEC.
002264  000007                DELAY2: .WORD   7               ;DELAY TIME FOR M-LOOP FUNCTION, 100 USEC.APPROX.
002266  000000                LOGDEV: .WORD   0               ;LOGICAL DEVICE NUMBER
002270  000000                PSTACK: .WORD   0               ;CONTAINS BASE LEVEL PROGRAM STACK POINTER
```

MISCELLANEOUS STORAGE

```
1755 002272  000000                INTFLG: .WORD   0               ;INTERRUPT RECEIVED FLAG BYTES.  ALLOCATION:
1756                                                               ;  LOW BYTE FOR "A" & HIGH BYTE FOR "B"
1757 002274  000000                INTWCH: .WORD   0               ;BYTE IS SET NON-ZERO WHEN HANDLER SHOULD BE
1758                                                               ;  WATCHING FOR INT'S.  ALLOCATION: SEE INTFLG
1759 002276  000000                ERRFLG: .WORD   0               ;ERROR FLAG
1760 002300  000000                REGNUM: .WORD   0               ;REGISTER NUMBER -- FOR PASSING ARG. TO "ERR#"
1761 002302  000000                FRSTIM: .WORD   0               ;FLAG=0 IF PROGRAM JUST LOADED
1762 002304  000000                FRSPAS: .WORD   0               ;FLAG=0 IF FIRST PASS AFTER LOAD
1763 002306  000000                STARES: .WORD   0               ;FLAG TO SHOW NO. OF PASSES SINCE STA OR RES
1764 002310  000000                DEVMAP: .WORD   0               ;BIT MAP OF ACTIVE DEVICES
1765 002312  000000                DEVPTR: .WORD   0               ;DEVICE MAP BIT POINTER
1766 002314  000000                CONSOL: .WORD   0               ;CONSOLE DEVICE FLAG -- NON-ZERO = NONE PRESENT
1767 002316  000000                PFLAG:  .WORD   0               ;MISC. PROGRAM FLAGS
1768                                    ;
1769                                    ; THE ABOVE WORD CONTAINS MISC. FLAGS WHICH CAN ONLY BE ACCESSED BY PATCHING.
1770                                    ; IT IS NOT INTENDED THAT THEY BE SET OR CLEARED EXCEPT UNDER VERY UNUSUAL
1771                                    ; CIRCUMSTANCES.  THEREFORE, THEY WILL NOT BE DOCUMENTED ANY OTHER PLACE
1772                                    ; EXCEPT RIGHT HERE.
1773                                    ;
1774                                    ; BIT 0 -- WHEN SET, THOSE TESTS WHICH DO A BUS RESET WILL NOT BE EXECUTE.
1775                                    ;             THIS WAS INPLEMENTED TO SAVE WEAR & TEAR ON THE RX01 IN THE
1776                                    ;             DEVELOPMENT SYSTEM WHILE DOING LONG TERM TESTING OF ALL OTHER
1777                                    ;             TESTS.
1778                                    ;
1779                                    ; BIT 1 -- CPU TYPE (NOT USED).
1780                                    ;
1781                                    ; BIT 2 -- CONTROLS PRINTING OF EXTENDED ERROR INFORMATION DURING "MOVING
1782                                    ;          INVERSIONS TEST" OF RAM.  NORMALLY ONLY ADDRESS, GOOD & BAD
1783                                    ;          DATA, AND XOR WILL BE PRINTED.  IF THIS BIT IS SET HOWEVER,
1784                                    ;          INFORMATION IDENTIFYING WHERE WITHIN THE ALGORITHM THE ERROR
1785                                    ;          WAS DETECTED IS REPORTED.  THE FOLLOWING ABBREVIATIONS ARE USED
1786                                    ;          IN THE HEADING:
1787                                    ;                  BIT --- IDENTIFIES THE INNERMOST LOOP.  WHICH BIT IS
1788                                    ;                          BEING INVERTED AT EACH LOCATION.  BITS ARE
1789                                    ;                          IDENTIFIED AS 0 THROUGH 7.
1790                                    ;                  DATA -- IDENTIFIES THE VALUE TO WHICH THE ABOVE BIT IS
1791                                    ;                          BEING SET (I.E. 0 OR 1).  IT IS FIRST READ AND
1792                                    ;                          CHECKED FOR EXPECTED CONTENTS; THEN THE BIT IS
1793                                    ;                          INVERTED TO THIS STATE (DATA) AND RE-WRITTEN;
1794                                    ;                          THEN IT IS AGAIN READ & CHECKED FOR THE NEW
1795                                    ;                          VALUE.
1796                                    ;                  SEQ --- INDICATES THE DIRECTION (FWD OR BKWD) THE TEST
1797                                    ;                          WAS SCANING THROUGH RAM WHEN THE ERROR OCCURED.
1798                                    ;                  LSB --- THIS IS THE LOGICAL LEAST SIGNIFICANT BIT OF THE
1799                                    ;                          RAM ADDRESS AS WE SCAN THROUGH MEMORY.  BY
1800                                    ;                          VARYING THIS, THE ALGORITHM GENERATES NON-SEQUEN-
1801                                    ;                          TIAL ADDRESSING OF RAM AND EFFECTS A MUCH MORE
1802                                    ;                          THOROUGH TEST OF MEMORY.
1803                                    ;
1804                                    ; BIT 3 -- ENABLES PRINTOUT OF THE MESSAGE WITHIN THE INIT CODE THAT TELLS
1805                                    ;          US HOW LONG IT TOOK (IN LOOPS WITHIN THIS CPU) TO PERFORM ONE
1806                                    ;          NO-OP FUNCTION OF THE MAINTAINENCE LOOP.
1807
1808
1809
```

CURRENT DEVICE PARAMETERS

```
                                .SBTTL  CURRENT DEVICE PARAMETERS

        160000                  $MPCSR  ==      160000          ;INITIAL ASSEMBLED IN CSR ADDRESS

002320                          MPCSR:                          ;POINTER TO THE DMV11 CSR'S
002320                          BSEL0:                          ;POINTER TO BSEL0
002320                          BSEL:                           ;ALTERNATE NAME FOR BSEL0
002320  160000                  SEL0:   .WORD   $MPCSR          ;POINTER TO SEL0
002322  160001                  BSEL1:  .WORD   $MPCSR+1        ;POINTER TO BSEL1
002324                          BSEL2:                          ;POINTER TO BSEL2
002324  160002                  SEL2:   .WORD   $MPCSR+2        ;POINTER TO SEL2
002326  160003                  BSEL3:  .WORD   $MPCSR+3        ;POINTER TO BSEL3
002330                          BSEL4:                          ;POINTER TO BSEL4
002330  160004                  SEL4:   .WORD   $MPCSR+4        ;POINTER TO SEL4
002332  160005                  BSEL5:  .WORD   $MPCSR+5        ;POINTER TO BSEL5
002334                          BSEL6:                          ;POINTER TO BSEL6
002334  160006                  SEL6:   .WORD   $MPCSR+6        ;POINTER TO SEL6
002336  160007                  BSEL7:  .WORD   $MPCSR+7        ;POINTER TO BSEL7
002340                          BSEL10:                         ;POINTER TO BSEL10
002340  160010                  SEL10:  .WORD   $MPCSR+10       ;POINTER TO SEL10
002342  160011                  BSEL11: .WORD   $MPCSR+11       ;POINTER TO BSEL11
002344                          BSEL12:                         ;POINTER TO BSEL12
002344  160012                  SEL12:  .WORD   $MPCSR+12       ;POINTER TO SEL12
002346  160013                  BSEL13: .WORD   $MPCSR+13       ;POINTER TO BSEL13
002350                          BSEL14:                         ;POINTER TO BSEL14
002350  160014                  SEL14:  .WORD   $MPCSR+14       ;POINTER TO SEL14
002352  160015                  BSEL15: .WORD   $MPCSR+15       ;POINTER TO BSEL15
002354                          BSEL16:                         ;POINTER TO BSEL16
002354  160016                  SEL16:  .WORD   $MPCSR+16       ;POINTER TO SEL16
002356  160017                  BSEL17: .WORD   $MPCSR+17       ;POINTER TO BSEL17

002360  000300                  MPIVEC: .WORD     300           ;DMV11 INPUT INTERRUPT VECTOR
002362  000304                  MPOVEC: .WORD     304           ;DMV11 OUTPUT INTERRUPT VECTOR
002364  000340                  MPRIOR: .WORD     340           ;DMV11 DEVICE PRIORITY
002366  000000                  BRDTYP: .WORD       0           ;0=M8064,1=M8053/V.35,2=M8053/EIA
002370  000000                  PT.CTL: .WORD       0           ;MISC. CONTROL FLAGS FROM P-TABLE

                                .SBTTL  GEN'L PURPOSE SCRATCH STORAGE

002372  000000                  REG0:   .WORD   0
002374  000000                  REG1:   .WORD   0
002376  000000                  REG2:   .WORD   0
002400  000000                  REG3:   .WORD   0
002402  000000                  REG4:   .WORD   0
002404  000000                  REG5:   .WORD   0
002406  000000                  REG6:   .WORD   0
002410  000000                  REG7:   .WORD   0
```

```
1859                                    .SBTTL  SCRATCH STORAGE FOR MESSAGE REPORTING
1860
1861 002412  000000                     TMP0:   .WORD   0
1862 002414  000000                     TMP1:   .WORD   0
1863 002416  000000                     TMP2:   .WORD   0
1864 002420  000000                     TMP3:   .WORD   0
1865 002422  000000                     TMP4:   .WORD   0
1866 002424  000000                     TMP5:   .WORD   0
1867 002426  000000                     TMP6:   .WORD   0
1868 002430  000000                     TMP7:   .WORD   0
1869 002432  000000                     TMP8:   .WORD   0
1870 002434  000000                     TMP9:   .WORD   0
1871 002436  000000                     TMPA:   .WORD   0
1872 002440  000000                     TMPB:   .WORD   0
1873 002442  000000                     TMPC:   .WORD   0
1874 002444  000000                     TMPD:   .WORD   0
1875 002446  000000                     TMPE:   .WORD   0
1876 002450  000000                     TMPF:   .WORD   0
1877 002452  000000                     NEWPC:  .WORD   0       ;SAVE LOCATION FOR A "PC" VALUE RESET
1878 002454  000000                     OLDSP:  .WORD   0       ;SAVE LOCATION FOR A STACK POINTER RESET VALUE
1879
1880
1881                                    .SBTTL  ***** DATA PATTERN B *****
1882
1883                                    .EVEN                           ;USAGE:
1884 002456  000025                     PATB:   .WORD   1$-.-2  ;# OF BYTES IN PATTERN.
1885 002460      125                            .BYTE   125
1886 002461      252                            .BYTE   252
1887 002462      000                            .BYTE   000
1888 002463      377                            .BYTE   377
1889 002464      001                            .BYTE   001
1890 002465      002                            .BYTE   002
1891 002466      004                            .BYTE   004
1892 002467      010                            .BYTE   010
1893 002470      020                            .BYTE   020
1894 002471      040                            .BYTE   040
1895 002472      100                            .BYTE   100
1896 002473      200                            .BYTE   200
1897 002474      376                            .BYTE   376
1898 002475      375                            .BYTE   375
1899 002476      373                            .BYTE   373
1900 002477      367                            .BYTE   367
1901 002500      357                            .BYTE   357
1902 002501      337                            .BYTE   337
1903 002502      277                            .BYTE   277
1904 002503      177                            .BYTE   177
1905 002504      000                            .BYTE   000
1906 002505                             1$:
```

***** DATA PATTERN F *****

```
1908                                    .SBTTL  ***** DATA PATTERN F *****
1909
1910                                    .EVEN
1911 002506  000045                     PATF:   <1$-.-2>/2
1912 002510  125252                             125252
1913 002512  052525                             052525
1914 002514  000000                             0
1915 002516  177777                             -1
1916 002520  000001                             1
1917 002522  000002                             2
1918 002524  000004                             4
1919 002526  000010                             10
1920 002530  000020                             20
1921 002532  000040                             40
1922 002534  000100                             100
1923 002536  000200                             200
1924 002540  000400                             400
1925 002542  001000                             1000
1926 002544  002000                             2000
1927 002546  004000                             4000
1928 002550  010000                             10000
1929 002552  020000                             20000
1930 002554  040000                             40000
1931 002556  100000                             100000
1932 002560  177776                             177776
1933 002562  177775                             177775
1934 002564  177773                             177773
1935 002566  177767                             177767
1936 002570  177757                             177757
1937 002572  177737                             177737
1938 002574  177677                             177677
1939 002576  177577                             177577
1940 002600  177377                             177377
1941 002602  176777                             176777
1942 002604  175777                             175777
1943 002606  173777                             173777
1944 002610  167777                             167777
1945 002612  157777                             157777
1946 002614  137777                             137777
1947 002616  077777                             077777
1948 002620  000000                             0
1949 002622                             1$:
```

***** DATA PATTERN RESULTS TABLE FOR MASTER CLEAR (RESFMC) ***

```
                                        .SBTTL  ***** DATA PATTERN RESULTS TABLE FOR MASTER CLEAR (RESFMC) *****
                                        .EVEN
002622  000                     BSELRS: .BYTE   000             ;BSEL0
002623  200                             .BYTE   200             ;BSEL1 -- "RUN" BIT SET
002624  000                             .BYTE   000             ;BSEL2
002625  000                             .BYTE   000             ;BSEL3
002626  033                             .BYTE   033             ;BSEL4 -- CODE FOR THE DMV-11
002627  000                             .BYTE   000             ;BSEL5
002630  305                             .BYTE   305             ;BSEL6 -- INDICATING VALID COMPLETION OF U-DIAG.
002631  000                             .BYTE   000             ;BSEL7


                                        .SBTTL  ***** DATA PATTERN OF NPR REG'S AFTER MASTER CLEAR ****

                                        ;       ALTHOUGH THE REGISTERS ARE ONLY 1 BYTE LONG, EACH TABLE ENTRY IS ONE
                                        ;       WORD LONG TO SIMPLIFY THE ERROR CHECKING & REPORTING.  THE HIGH BYTE
                                        ;       OF EACH ENTRY MUST BE LEFT AT ZERO OR THE TESTING & PRINTING WILL BE
                                        ;       IN ERROR!

                                        .EVEN
002632  000004                  NPRMCR: .WORD   DMVPU    ;ONLY THE "POWER UP" IS SET (BY MICRO-DIAG.)
002634  000000                          .WORD   0       ;"DATA HI"
002636  000000                          .WORD   0       ;"DATA LO"
002640  000000                          .WORD   0       ;"OUT ADDR. EXTENDED"
002642  000000                          .WORD   0       ;"OUT ADDR. HI"
002644  000000                          .WORD   0       ;"OUT ADDR. LO"
002646  000000                          .WORD   0       ;"IN ADDR. EXTENDED"
002650  000000                          .WORD   0       ;"IN ADDR. HI"
002652  000000                          .WORD   0       ;"IN ADDR. LO"
```

```
1982                                    .SBTTL  DATA BUFFER AREAS
1983
1984 002654                             BUFAREA:        .BLKB   256.
1985
1986
1987                                    ;  THIS BUFFER HAS SOME ALTERNATE USES TOO.  THE FOLLOWING LABELS ARE PROVIDED
1988                                    ;  FOR THOSE USAGES.
1989
1990            003054                  W0 = BUFAREA+128.        ;THIS WORD TABLE STARTS IN THE MIDDLE OF "BUFAREA"
1991            003056                  W1 = W0+2                ;AND IS USED BY "ERR6" FOR PRINTING BYTES
1992            003060                  W2 = W1+2
1993            003062                  W3 = W2+2
1994            003064                  W4 = W3+2
1995            003066                  W5 = W4+2
1996            003070                  W6 = W5+2
1997            003072                  W7 = W6+2
1998            003074                  W8 = W7+2
1999            003076                  W9 = W8+2
2000            003100                  WA = W9+2
2001            003102                  WB = WA+2
2002            003104                  WC = WB+2
2003            003106                  WD = WC+2
2004            003110                  WE = WD+2
2005            003112                  WF = WE+2
2006
2007            002654                  BT1 = BUFAREA            ;BYTE TABLE # 1
2008            002740                  BT2 = BUFAREA+64         ;BYTE TABLE # 2
```

GLOBAL TEXT SECTION

```
2010                                    .SBTTL GLOBAL TEXT SECTION
2011
2012                                    ;*****************************************************************************
2013                                    ;*      THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
2014                                    ;*      MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
2015                                    ;*      MORE THAN ONE TEST.
2016                                    ;*****************************************************************************
2017
2018                                    ;++*************************************************************************
2019                                    ;* NAMES OF DEVICES SUPPORTED BY PROGRAM
2020                                    ;--*************************************************************************
2021 003254                                     DEVTYP  <M8053 OR M8064>
     003254                                                                L$DVTYP::
     003254     115     070     060                                              .ASCIZ  %M8053 OR M8064%
     003257     065     063     040
     003262     117     122     040
     003265     115     070     060
     003270     066     064     000

                                                                          .EVEN
2022
2023                                    ;++*************************************************************************
2024                                    ;* TITLE OF PROGRAM
2025                                    ;--*************************************************************************
2026
2027           000012                   .RADIX  10.
2028 003274                             DESCRIPT    <DMV-11 U-CONTRL LOGIC DIAG - PART 2 OF 2>
     003274                                                                L$DESC::
     003274      104     115     126                                              .ASCIZ  /DMV-11 U-CONTRL LOG
IC DIAG - PART 2 OF 2/
     003277      055     061     061
     003302      040     125     055
     003305      103     117     116
     003310      124     122     114
     003313      040     114     117
     003316      107     111     103
     003321      040     104     111
     003324      101     107     040
     003327      055     040     120
     003332      101     122     124
     003335      040     062     040
     003340      117     106     040
     003343      062     000

                                                                          .EVEN
2029           000010                   .RADIX  8.
2030
```

```
                        .SBTTL GLOBAL SUBROUTINES

                        ;////////////////////////////////////////////////////////////////////////////
                        ;/      THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
                        ;////////////////////////////////////////////////////////////////////////////

                        ;+*************************************************************************
                        .SBTTL  MASCLR -  MASTER CLEAR SUBROUTINE
                        ;
                        ; FUNCTION:
                        ;
                        ;  THIS  SUBROUTINE  FORCES THE 6502 MICROPROCESSOR TO EXECUTE A MINI 17 PART
                        ;  DIAGNOSTIC OF THE MICRO-PROCESSOR INSTRUCTION SET, RAM DATA AND ADDRESSING
                        ;  VALIDITY,   AND   A  ROM  CRC  TEST.  THE  CLEAR  SUBROUTINE  EXECUTES  IN
                        ;  APPROXIMATELY  500  HUNDRED(S)  MILLISECOND. THIS SUBROUTINE WILL SEND THE
                        ;  MASTER  CLEAR  COMMAND  AND  DELAY FOR APPROX. 500 MSEC. AT WHICH PRINT IN
                        ;  TIME,  THE  STATE  OF  THE  CSR  REGISTERS  IS  TESTED.  IF ANY ONE OF THE
                        ;  REGISTERS CONTAINS ANYTHING THAT IS NOT EXPECTED, AN ERROR IS QUEUE UP AND
                        ;  THE CARRY BIT IS SET. ELSE, THE CARRY BIT IS CLEARED.
                        ;
                        ; CALLING SEQUENCE:
                        ;
                        ;       JSR     PC,MASCLR
                        ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
                        ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
                        ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
                        ;
                        ; N$:   <RESUMPTION OF NORMAL PROCESSING>
                        ;
                        ;-*************************************************************************
                        ;
003346  010146          MASCLR: MOV     R1,-(SP)        ; SAVE REGISTER ONE
003350  112777  000300  176744          MOVB    @RUN!MCLR,@BSEL1 ;SET BOTH THE RUN AND MASTER CLEAR BITS
                                                        ;TO INITIATE THE MICRODIAGNOSTIC

                                        ;NOW DELAY LONG ENOUGH FOR THE MICRODIAGNOSTIC TO COMPLETE

003356  013701  002262                  MOV     DELAY1,R1       ;INITIALIZE THE LOOP COUNTER FOR DELAY LOOP
003362  001402                  2$:     BEQ     1$              ; EXIT DELAY LOOP IF THE TIME HAS EXPIRED
003364  005301                          DEC     R1              ;  ELSE, DECREMENT THE LOOP COUNTER AND
003366  000775                          BR      2$              ;  CONTINUE TO LOOP.
003370                          1$:                             ; TIME-UP!
003370  132777  000200  176724          BITB    @RUN,@BSEL1     ;CHECK THE RUN BIT --
003376  001410                          BEQ     3$              ;IF NOT SET, GO REPORT THE ERROR

                                        ;IF THE RUN BIT IS SET, MICRODIAGNOSTICS ARE COMPLETE.
                                        ;CHECK IF ALL MICRODIAGNOSTICS PASSED.

003400  127737  176730  002630  4$:     CMPB    @BSEL6,BSELRS+6 ;THIS CHECKS THE BYTE IN B-SELECT 6 FOR THE
                                                        ;VALID MICRODIAGNOSTIC COMPLETION CODE.
003406  001004                          BNE     3$              ;IF BAD, GO REPORT ERROR

003410  127737  176714  002626          CMPB    @BSEL4,BSELRS+4 ;ELSE, CHECK FOR THE VALID CODE FOR A DMV-11
003416  001420                          BEQ     6$              ;IF THIS TOO IS CORRECT THEN NO ERROR EXISTS
                                                        ;ELSE, FALL INTO THE ERROR REPORTING CODE
```

```
2095 003420  004737  004232          3$:     JSR     PC,GETBSR       ;GET THE BSEL REGISTERS FOR DUMPING
2096 003424                                  GTDF    20$,ERR3        ;MASTER CLEAR ERROR
                                             ;       QUEUE "DEVICE FATAL" ERROR # 1
     003424  012737  000001  002202                                                  MOV     #T.EDF,ERRTYP
     003432  012737  000001  002204                                                  MOV     #1,ERRNBR
     003440  012737  003466  002206                                                  MOV     #20$,ERRMSG
     003446  012737  006220  002210                                                  MOV     #ERR3,ERRBLK
2097 003454  000261                          SEC                     ;INDICATE TO THE CALLING ROUTINE THAT
2098 003456  000401                          BR      7$              ;   AN ERROR WAS DETECTED
2099
2100 003460  000241                  6$:     CLC                     ;CLEAR THE CARRY BIT TO INDICATE NO ERROR
2101 003462  012601                  7$:     MOV     (SP)+,R1        ;RESTORE REGISTER ONE
2102 003464  000207                          RTS     PC              ; RETURN TO THE CALLER
2103                                 .NLIST  BEX
2104 003466     115     101     123  20$:    .ASCIZ  /MASTER CLEAR FAILURE/
2105                                 .LIST   BEX
2106                                 .EVEN
```

```
                                .SBTTL  M-LOOP -- MSTCLR -- MASTER CLEAR & ENTER M-LOOP
                                ;++*************************************************************************
                                ;  MSTCLR -- MASTER CLEAR & ENTER M-LOOP
                                ;
                                ;  CALLING SEQUENCE:
                                ;
                                ;       JSR     PC,MSTCLR
                                ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
                                ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
                                ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
                                ;
                                ; N$:   <RESUMPTION OF NORMAL PROCESSING>
                                ;
                                ;--*************************************************************************

003514  012777  140400  176576  MSTCLR: MOV     #<RUN!MCLR!MREQ>*256.,@SELO  ;INITIATE M-LOOP

003522  010346                          MOV     R3,-(SP)
003524  012703  000014                  MOV     #12.,R3         ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
003530  077301                  1$:     SOB     R3,1$
003532  012603                          MOV     (SP)+,R3

003534  132777  000200  176562          BITB    #MRDY,@BSEL2    ;DID THE M-LOOP FINISH
003542  001023                          BNE     5$              ;YES,  GOOD.  RETURN
003544  004737  004374                  JSR     PC,GETWSR       ;GET BYTE SELECT REGISTERS
003550  012737  000301  002254          MOV     #RUN!MCLR!MREQ,GDATA    ;IDENTIFY REQUESTED FUNCTION
003556                                  GTDF    EM3,ERR4        ;"MRDY" TIMEOUT
                                        ;       QUEUE "DEVICE FATAL" ERROR # 2
003556  012737  000001  002202                                          MOV     #T.EDF,ERRTYP
003564  012737  000002  002204                                          MOV     #2,ERRNBR
003572  012737  016115  002206                                          MOV     #EM3,ERRMSG
003600  012737  006232  002210                                          MOV     #ERR4,ERRBLK
003606  000261                          SEC                     ;SET CARRY TO INDICATE ERROR
003610  000401                          BR      9$              ;EXIT WITH THE "ERROR" FLAG (CARRY BIT) SET
003612  000241                  5$:     CLC                     ;CLEAR C BIT FOR NO ERRORS
003614  000207                  9$:     RTS     PC              ;RETURN
```

```
                                .SBTTL  M-LOOP -- READ
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ; READ - READ THE SPECIFIED ADDRESS WITHIN THE DMV-11
                                ;
                                ;  CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,READ
                                ;       .WORD   <ADDRESS OF REGISTER WITHIN DMV-11>
                                ;       .WORD   <DESTINATION ADDRESS WITHIN LSI-11>
                                ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
                                ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
                                ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
                                ;
                                ; N$:   <RESUMPTION OF NORMAL PROCESSING>
                                ;
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

003616  012577  176506        READ:   MOV     (R5)+,@SEL4     ;SETUP SOURCE POINTER
003622  112777  000001  176474        MOVB    #REDLOC,@BSEL2  ;TELL M-LOOP TO GIVE US THE REQUESTED DATA

003630  010346                        MOV     R3,-(SP)
003632  012703  000032                MOV     #26.,R3         ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
003636  077301                1$:     SOB     R3,1$
003640  012603                        MOV     (SP)+,R3

003642  132777  000200  176454        BITB    @MRDY,@BSEL2    ;DID THE M-LOOP FINISH
003650  001023                        BNE     5$              ;YES,  GOOD.  RETURN

003652  004737  004374                JSR     PC,GETWSR       ;GET BYTE SELECT REGISTERS
003656  012737  000001  002250        MOV     #REDLOC,GDATA   ;IDENTIFY REQUESTED FUNCTION
003664                                GTDF    EM4,ERR4        ;"MRDY" TIMEOUT
                                                        ;       QUEUE "DEVICE FATAL" ERROR # 3
003664  012737  000001  002[illegible]                                MOV     #T.EDF,ERRTYP
003672  012737  000003  002[illegible]                                MOV     #3,ERRNBR
003700  012737  016135  002206                                        MOV     #EM4,ERRMSG
003706  012737  006232  002210                                        MOV     #ERR4,ERRBLK
003714  000261                        SEC                     ;INDICATE AN ERROR HAS BEEN STACKED
003716  000401                        BR      6$              ;RETURN WITH THAT INDICATION

003720  000241                5$:     CLC                     ;INDICATE "NO ERROR"
003722  117735  176406        6$:     MOVB    @BSEL6,@(R5)+   ;PUT DATA WHERE CALLER WANTS IT
003726  000205                        RTS     R5              ;RETURN
```

```
                                .SBTTL  M-LOOP -- READ IMMEDIATE
                                ;++*************************************************************************
                                ; READI - READ IMMEDIATE THE SPECIFIED ADDRESS WITHIN THE DMV-11
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,READI
                                ;       .WORD   <ADDRESS OF REGISTER WITHIN DMV-11>
                                ;       .WORD   <DESTINATION -- CONTENTS OF REG. IS PUT HERE>
                                ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
                                ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
                                ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
                                ;
                                ; N$:   <RESUMPTION OF NORMAL PROCESSING>
                                ;
                                ;--*************************************************************************

003730                          READI:
003730  012577  176374                  MOV     (R5)+,@SEL4     ;SETUP SOURCE POINTER
003734  112777  000001  176362          MOVB    @REDLOC,@BSEL2  ;TELL M-LOOP TO GIVE US THE REQUESTED DATA

003742  010346                          MOV     R3,-(SP)
003744  012703  000015                  MOV     #13.,R3         ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
003750  077301                  1$:     SOB     R3,1$
003752  012603                          MOV     (SP)+,R3

003754  132777  000200  176342          BITB    @MRDY,@BSEL2    ;DID THE M-LOOP FINISH
003762  001023                          BNE     5$              ;YES.  GOOD.  RETURN

003764  004737  004374                  JSR     PC,GETWSR       ;GET BYTE SELECT REGISTERS
003770  012737  000001  002254          MOV     #REDLOC,GDATA   ;IDENTIFY REQUESTED FUNCTION
003776                                  GTDF    EM4,ERR4        ;"MRDY" TIMEOUT
                                        ;       QUEUE "DEVICE FATAL" ERROR # 4
003776  012737  000001  002202                                          MOV     #T.EDF,ERRTYP
004004  012737  000004  002204                                          MOV     #4,ERRNBR
004012  012737  016135  002206                                          MOV     #EM4,ERRMSG
004020  012737  006232  002210                                          MOV     #ERR4,ERRBLK
004026  000261                          SEC                     ;INDICATE AN ERROR HAS BEEN STACKED
004030  000401                          BR      6$              ;RETURN WITH THAT INDICATION

004032  000241                  5$:     CLC                     ;INDICATE "NO ERROR"
004034  017725  176274          6$:     MOV     @SEL6,(R5)+     ;PUT DATA WHERE CALLER WANTS IT
004040  000205                          RTS     R5              ;RETURN
```

```
                                .SBTTL  M-LOOP -- WRITE
                                ;**************************************************************************
                                ; WRITE - WRITE THE SPECIFIED DATA INTO THE SPECIFIED DMV-11 ADDRESS
                                ;
                                ;  CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,WRITE
                                ;       .WORD   <ADDRESS OF REGISTER WITHIN DMV-11>
                                ;       .WORD   <ADDRESS OF DATA BYTE>
                                ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
                                ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
                                ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
                                ;
                                ; N$:   <RESUMPTION OF NORMAL PROCESSING>
                                ;
                                ;--**************************************************  **********************

004042  012577  176262         WRITE:  MOV     (R5)+,@SEL4     ;SETUP SOURCE POINTER
004046  113577  176262                 MOVB    @(R5)+,@SEL6    ;MAKE DATA AVAILABLE TO M-LOOP
004052  000404                         BR      MLWRI           ;THE REST OF THIS ROUTINE IS THE SAME AS "WRITEI"
```

```
2238                                    .SBTTL  M-LOOP -- WRITE IMMEDIATE
2239                                    ;**************************************************************************
2240                                    ; WRITEI - WRITE IMMEDIATE THE SPECIFIED DATA INTO THE SPECIFIED DMV-11 ADDRESS
2241                                    ;
2242                                    ; CALLING SEQUENCE:
2243                                    ;
2244                                    ;       JSR     R5,WRITEI
2245                                    ;       .WORD   <ADDRESS OF REGISTER WITHIN DMV-11>
2246                                    ;       .WORD   <DATA FIELD -- DATA TO BE WRITTEN IN DMV-11>
2247                                    ;       BCC     N$              ;IF NO ERROR OCCURED, PROCEED WITH ROUTINE
2248                                    ;       ERROR                   ;AN ERROR MESSAGE HAS BEEN STACKED:  PRINT IT
2249                                    ;       <ANY OTHER SPECIAL ERROR PROCESSING MAY BE DONE HERE  (I.E. CKLOOP)>
2250                                    ;
2251                                    ; N$:   <RESUMPTION OF NORMAL PROCESSING>
2252                                    ;
2253                                    ;--**************************************************************************
2254
2255 004054                             WRITEI:
2256 004054  012577  176250                        MOV     (R5)+,@SEL4     ;SETUP SOURCE POINTER
2257 004060  012577  176250                        MOV     (R5)+,@SEL6     ;MAKE DATA AVAILABLE TO M-LOOP
2258 004064  112777  000002  176232   MLWRI:      MOVB    #WRILOC,@BSEL2  ;TELL M-LOOP TO WRITE THE DATA
2259
2260 004072  010346                             MOV     R3,-(SP)
2261 004074  012703  000050                        MOV     #40.,R3         ;WAIT FOR THE M-LOOP TO FINISH THE OPERATION
2262 004100  077301                        1$:     SOB     R3,1$
2263 004102  012603                             MOV     (SP)+,R3
2264
2265 004104  132777  000200  176212             BITB    #MRDY,@BSEL2    ;DID THE M-LOOP FINISH
2266 004112  001023                             BNE     5$              ;YES,  GOOD.  RETURN
2267 004114  004737  004374                        JSR     PC,GETWSR       ;GET BYTE SELECT REGISTERS
2268 004120  012737  000002  002254             MOV     #WRILOC,GDATA   ;IDENTIFY REQUESTED FUNCTION
2269 004126                                     GTDF    EM4,ERR4        ;"MRDY" TIMEOUT
                                                                        ;       QUEUE "DEVICE FATAL" ERROR # 5
     004126  012737  000001  002202                                                        MOV     #T.EDF,ERRTYP
     004134  012737  000005  002204                                                        MOV     #5,ERRNBR
     004142  012737  016135  002206                                                        MOV     #EM4,ERRMSG
     004150  012737  006232  002210                                                        MOV     #ERR4,ERRBLK
2270 004156  000261                             SEC                     ;INDICATE AN ERROR HAS BEEN STACKED
2271 004160  000401                             BR      6$              ;RETURN WITH THAT INDICATION
2272
2273 004162  000241                        5$:     CLC                     ;INDICATE "NO ERROR"
2274 004164  000205                        6$:     RTS     R5              ;RETURN
```

```
                                .SBTTL  M-LOOP --- BLOCK MOVE (11 CPU ==> DMV-11)
                                ;*******************************************************************************
                                ;
                                ; MOVE A BLOCK OF BYTES FROM THE 11 CPU'S MEMORY TO THE DMV-11'S RAM
                                ;
                                ;       THE DESTINATION ADDRESS IS ALWAYS THE SAME -- 77 (OCT) OR 003F (HEX)
                                ;
                                ;       CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,MOVLTD
                                ;       .WORD   <SOURCE ADDRESS OF DATA>
                                ;       .WORD   <# OF BYTE TO BE MOVED>
                                ;
                                ;*******************************************************************************
004166  010146                  MOVLTD: MOV     R1,-(SP)        ;SAVE THE REGISTER WE'LL BE A NEED'N

004170  012737  000077  004210          MOV     #77,10$         ;DESTINATION ADDRESS IS ALWAYS THE SAME
004176  012537  004212                  MOV     (R5)+,11$       ;SETUP THE SOURCE ADDRESS
004202  012501                          MOV     (R5)+,R1        ;INITIALIZE THE BYTE COUNT

004204                          5$:
004204  004537  004042                  JSR     R5,WRITE        ;WRITE ONE BYTE INTO THE DMV-11'S RAM
004210  000077                  10$:    77                      ;THE RAM'S LOCATION
004212  000000                  11$:    0                       ;THE DATA BYTE'S LOCATION
004214  005237  004210                  INC     10$             ;POINT TO NEXT RAM BYTE
004220  005237  004212                  INC     11$             ;POINT TO NEXT DATA BYTE
004224  077111                          SOB     R1,5$           ;IF NOT DONE, LOOP
                                                                ;ELSE, CLEAN-UP AND RETURN
004226  012601                          MOV     (SP)+,R1        ;RESTORE R1
004230  000205                          RTS     R5              ;  & RETURN
```

```
                                .SBTTL  GETBSR -- GET BYTE SELECT REGISTERS

                                ;**************************************************************************
                                ;
                                ;       GET THE CONTENTS OF ALL CONTROL AND STATUS REGISTERS
                                ;
                                ;       FUNCTION - THIS SUBROUTINE COLLECTS THE CONTENTS OF THE
                                ;               BYTE SELECT REGISTERS FOR THE PURPOSE OF DISPLAY.
                                ;
                                ;       ENTRY CONDITIONS - NONE         ##   #     ####     #     ##   #
                                ;                                       # #  #     #        # #    # # #
                                ;        EXIT CONDITIONS - NONE         #     #      ##    #   #  #  ##
                                ;                                       # #  #     #      #####  #    #
                                ;    REGISTERS DESTROYED - NONE         ##   ####   ####  #   #  #    #
                                ;
                                ;--**************************************************************************

004232  117737  176062  002212  GETBSR: MOVB    @#SEL0,BSR0     ;PUT THE CURRENT CSR VALUES INTO THE PRINT-OUT
004240  117737  176056  002214          MOVB    @#SEL1,BSR1     ;TABLE
004246  117737  176052  002216          MOVB    @#SEL2,BSR2
004254  117737  176046  002220          MOVB    @#SEL3,BSR3
004262  117737  176042  002222          MOVB    @#SEL4,BSR4
004270  117737  176036  002224          MOVB    @#SEL5,BSR5
004276  117737  176032  002226          MOVB    @#SEL6,BSR6
004304  117737  176026  002230          MOVB    @#SEL7,BSR7
004312  117737  176022  002232          MOVB    @#SEL10,BSR10
004320  117737  176016  002234          MOVB    @#SEL11,BSR11
004326  117737  176012  002236          MOVB    @#SEL12,BSR12
004334  117737  176006  002240          MOVB    @#SEL13,BSR13
004342  117737  176002  002242          MOVB    @#SEL14,BSR14
004350  117737  175776  002244          MOVB    @#SEL15,BSR15
004356  117737  175772  002246          MOVB    @#SEL16,BSR16
004364  117737  175766  002250          MOVB    @#SEL17,BSR17
004372  000207                          RTS     PC              ;RETURN TO CALLER

                                .SBTTL  GETWSR -- GET WORD SELECT REGISTERS
                                ; "WORD" VERSION OF ABOVE SUBROUTINE

004374  017737  175720  002212  GETWSR: MOV     @#SEL0,WSR0     ;MOVE THE 4 WORD REGISTERS TO THE OTHERWISE
004402  017737  175716  002214          MOV     @#SEL2,WSR2     ;BYTE TABLE
004410  017737  175714  002216          MOV     @#SEL4,WSR4
004416  017737  175712  002220          MOV     @#SEL6,WSR6
004424  017737  175710  002222          MOV     @#SEL10,WSR10
004432  017737  175706  002224          MOV     @#SEL12,WSR12
004440  017737  175704  002226          MOV     @#SEL14,WSR14
004446  017737  175702  002230          MOV     @#SEL16,WSR16
004454  000207                          RTS     PC              ;RETURN TO CALLER
```

```
                        .SBTTL  .INITT1 -- INITIALIZE TIMER # 1

                        ;*****************************************************************************
                        ;* INITT1 - INITIALIZE TIMER # 1
                        ;*
                        ;*      CALLING SEQUENCE:
                        ;*
                        ;*          JSR     R5,INITT1
                        ;*          .WORD   <VALUE LOADED INTO THE T1 LATCH @ T1LL & T1LH>
                        ;*          .WORD   <BITS 6 & 7 WILL BE LOADED INTO "ACR", BIT 5 WILL BE
                        ;*                   USED TO SET OR CLEAR BIT 6 ("T1") OF THE INTERRUPT
                        ;*                   ENABLE REGISTER ("IER")>
                        ;*
                        ;*
                        ;* SEQUENCE OF EVENTS HEREIN:
                        ;*
                        ;*      SET THE VIA'S INTERRUPT ENABLE REGISTER ("IER")
                        ;*
                        ;*      SET THE VIA'S "ACR"
                        ;*
                        ;*      SET T1L-L (ADDR 06)
                        ;*
                        ;*      SET T1L-H (ADDR 07)
                        ;*
                        ;*      RETURN WITHOUT ANY ERROR CHECKING
                        ;*
                        ;*****************************************************************************

004456  010146          INITT1: MOV     R1,-(SP)        ;SAVE THE REGISTER WE WILL BE USING
004460  112537  002427          MOVB    (R5)+,TMP6+1    ;SETUP VALUES TO BE LOADED INTO THE LATCHES
004464  112537  002431          MOVB    (R5)+,TMP7+1
004470  111537  002441          MOVB    (R5),TMPB+1     ;GET & PROCESS BITS FOR ACR 6 & 7
004474  142737  177477  002441  BICB    #^C<BIT6+BIT7>,TMPB+1 ;EXTRACT BITS 6 & 7 & SAVE THEM FOR LATER
004502  012501                  MOV     (R5)+,R1        ;NOW, GET THE BIT TO BE USED IN SETTING OR
                                                        ;CLEARING BIT 6 OF "IER"

                        ; THE PASSED BIT IS IN THE WRONG POSITION BUT, IT SHOULD CONTROL THE OPERATION.
                        ; WE KNOW WE ARE SETTING OR CLEARING BIT 6 - THUS, THE PASSED BIT WILL BECOME
                        ; THE CONTROLLING BIT 7 AND WE WILL "OR" IN THE BIT WE WISH TO BE CONTROLLED
                        ; (BIT 6).

004504  106301                  ASLB    R1              ;THIS PUTS THE PASSED BIT INTO BIT 6.
004506  042701  177677          BIC     #^C<BIT6>,R1    ;WHILE HERE, CLEAR ALL OTHER BITS AND
004512  140177  175610          BICB    R1,@#SEL3       ;CLEAR THE INTERRUPT FLAG IN THE SELECT REG.
004516  106301                  ASLB    R1              ;NOW THE BIT IS IN THE CONTROLLING POSITION
004520  052701  000100          BIS     #BIT6,R1        ;SET BIT 6
004524  110137  002447          MOVB    R1,TMPE+1       ;THE CALL WILL NOW WRITE THE APPROPRIATE VALUE

004530  004537  004042          JSR     R5,WRITE        ;WRITE TO
004534  120016                  IENR                    ;THE VIA'S IER
004536  002447                  TMPE+1                  ;INTERRUPT ENABLE/DISABLE INFORMATION
004540  103431                  BCS     63$             ;EXIT ON ERROR

004542  004537  003616          JSR     R5,READ         ;READ THE CURRENT SETTING OF
004546  120013                  ACR                     ;THE VIA'S ACR
004550  002440                  TMPB
004552  103424                  BCS     63$             ;EXIT ON ERROR
```

```

004554  013701  002440                  MOV     TMPB,R1                 ;GET THAT VALUE
004560  042701  177477                  BIC     #^C<BIT6+BIT7>,R1       ;CLEAR BITS 6 & 7
004564  150137  002441                  BISB    R1,TMPB+1               ;ADD CURRENT BITS 0 --> 5 TO NEW BITS 6 & 7

004570  004537  004042                  JSR     R5,WRITE                ;WRITE THE NEW REGISTER SETTING TO VIA'S ACR
004574  120013                          ACR
004576  002441                          TMPB+1
004600  103411                          BCS     63$                     ;EXIT ON ERROR

004602  004537  004042                  JSR     R5,WRITE                ;WRITE TO
004606  120006                          T1LL                            ;LOW ORDER LATCH REGISTER (T1L-L)
004610  002427                          TMP6+1                          ;THE VALUE PASSED
004612  103404                          BCS     63$                     ;EXIT ON ERROR

004614  004537  004042                  JSR     R5,WRITE                ;WRITE TO
004620  120007                          T1LH                            ;HIGH ORDER LATCH REGESTER (T1L-H)
004622  002431                          TMP7+1                          ;THE VALUE PASSED

                                ; DON'T WAIT AROUND FOR ANYTHING TO HAPPEN -- JUST (JEST) RETURN!

004624  012601                  63$:    MOV     (SP)+,R1                ;BUT FIRST RESTORE R1
004626  000205                          RTS     R5                      ;THEN RETURN
```

```
                        .SBTTL  .INITT2 -- INITIALIZE TIMER # 2

                        ;**********************************************************************
                        ;* INITT2 - INITIALIZE TIMER # 2
                        ;*
                        ;*       CALLING SEQUENCE:
                        ;*
                        ;*             JSR     R5,INITT2
                        ;*             .WORD   <VALUE LOADED INTO "T2L-L" & "T2C-H">
                        ;*             .BYTE   <BIT 5 WILL BE LOADED INTO "ACR", BIT 4 WILL BE USED
                        ;*                     TO SET OR CLEAR BIT 5 ("T2") OF THE INTERRUPT ENABLE
                        ;*                     REGISTER ("IER")>
                        ;*             .BYTE   <UNUSED>
                        ;*
                        ;*
                        ;* SEQUENCE OF EVENTS HEREIN:
                        ;*
                        ;*     SET THE VIA'S INTERRUPT ENABLE REGISTER ("IER")
                        ;*
                        ;*     SET THE VIA'S "ACR"
                        ;*
                        ;*     SET T2L-L (ADDR 08)
                        ;*
                        ;*     SET T2C-H (ADDR 09)
                        ;*
                        ;*     RETURN WITHOUT ANY ERROR CHECKING
                        ;*
                        ;**********************************************************************

004630  010146          INITT2: MOV     R1,-(SP)        ;SAVE THE REGISTER WE WILL BE USING
004632  112537  002433          MOVB    (R5)+,TMP8+1    ;SETUP VALUES TO BE WRITTEN INTO COUNTER
004636  112537  002435          MOVB    (R5)+,TMP9+1
004642  111537  002441          MOVB    (R5),TMP8+1     ;GET & PROCESS BIT FOR ACR 5
004646  142737  177737  002441  BICB    @#CBIT5,TMP8+1
004654  012501                  MOV     (R5)+,R1        ;NOW, GET THE BIT TO BE USED IN SETTING OR
                                                        ;CLEARING BIT 5 OF "IER"

                        ; THE PASSED BIT IS IN THE WRONG POSITION BUT, IT SHOULD CONTROL THE OPERATION.
                        ; WELL, WE KNOW WE ARE SETTING OR CLEARING BIT 5.  THUS, THE PASSED BIT WILL
                        ; BECOME THE CONTROLLING BIT 7 AND WE'LL "OR" IN THE BIT WE WISH TO BE
                        ; CONTROLLED (BIT 5).

004656  106301                  ASLB    R1              ;THE PASSED BIT IS NOW IN POSITION TO
004660  042701  177737          BIC     @#CBIT5,R1      ;CLEAR ALL UNWANTED BITS AND
004664  140177  175436          BICB    R1,@#SEL3       ;CLEAR THE INT. FLAG IN THE SELECT REGISTER
004670  106301                  ASLB    R1              ;NOW PUT THE BIT INTO THE CONTROL POSITION
004672  106301                  ASLB    R1
004674  052701  000040          BIS     @#BIT5,R1       ;THEN SET BIT 5
004700  110137  002447          MOVB    R1,TMPE+1       ;THE CALL WILL NOW WRITE THE APPROPRIATE VALUE

004704  004537  004042          JSR     R5,WRITE        ;WRITE TO
004710  120016                  IENR                    ;THE VIA'S IER
004712  002447                  TMPE+1                  ;INTERRUPT ENABLE/DISABLE INFORMATION
004714  103431                  BCS     63$             ;EXIT ON ERROR

004716  004537  003616          JSR     R5,READ         ;READ THE CURRENT SETTING OF
004722  120013                  ACR                     ;THE VIA'S ACR
```

```
.INITT2 -  INITIALIZE TIMER # 2

 2495 004724  002440                    TMPB
 2496 004726  103424                    BCS     63$             ;EXIT ON ERROR
 2497
 2498 004730  113701  002440            MOVB    TMPB,R1         ;GET THAT VALUE
 2499 004734  042701  000040            BIC     #BIT5,R1        ;CLEAR THE CURRENT SETTING OF BIT 5
 2500 004740  150137  002441            BISB    R1,TMPB+1       ;SET REMAINING BITS IN THE VALUE TO BE WRITTEN
 2501
 2502 004744  004537  004042            JSR     R5,WRITE        ;WRITE TO
 2503 004750  120013                    ACR                     ;THE VIA'S ACR
 2504 004752  002441                    TMPB+1
 2505 004754  103411                    BCS     63$             ;EXIT ON ERROR
 2506
 2507 004756  004537  004042            JSR     R5,WRITE        ;WRITE TO
 2508 004762  120010                    T2LL                    ;LOW ORDER LATCH & COUNTER (T2L-L)
 2509 004764  002433                    TMP8+1                  ;THE PASSED VALUE
 2510 004766  103404                    BCS     63$             ;EXIT ON ERROR
 2511
 2512 004770  004537  004042            JSR     R5,WRITE        ;WRITE TO
 2513 004774  120011                    T2CH                    ;HIGH ORDER COUNTER (T2C-H) <ALSO STARTS CTR>
 2514 004776  002435                    TMP9+1                  ;THE PASSED VALUE
 2515
 2516                           ; DON'T WAIT AROUND FOR ANYTHING TO HAPPEN -- JUST (JEST) RETURN!
 2517
 2518 005000  012601            63$:    MOV     (SP)+,R1        ;BUT FIRST RESTORE R1
 2519 005002  000205                    RTS     R5              ;THEN RETURN
 2520
```

MOVSW -- MOVE A STRING OF WORDS

```
                                .SBTTL  MOVSW -- MOVE A STRING OF WORDS
                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ; MOVSW -- MOVE A STRING OF WORDS
                                ;
                                ;       CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,MOVSW
                                ;       .WORD   <ADDRESS OF SOURCE STRING>
                                ;       .WORD   <ADDRESS OF DESTINATION STRING>
                                ;       .WORD   <# OF WORDS TO MOVE>
                                ;
                                ;---++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

005004  010146          MOVSW:  MOV     R1,-(SP)        ;SAVE THE REGISTERS WE'LL BE USING
005006  010246                  MOV     R2,-(SP)
005010  010346                  MOV     R3,-(SP)

005012  012501                  MOV     (R5)+,R1        ;INITIALIZE SOURCE POINTER
005014  012502                  MOV     (R5)+,R2        ;       DESTINATION POINTER
005016  012503                  MOV     (R5)+,R3        ;       COUNTER

005020  012122          1$:     MOV     (R1)+,(R2)+     ;MOVE IN 1 WORD OF DATA
005022  077302                  SOB     R3,1$           ;IF MORE DATA, LOOP
                                                        ;ELSE, RESTORE REGISTERS AND RETURN

005024  012603                  MOV     (SP)+,R3        ;RESTORE REGISTERS
005026  012602                  MOV     (SP)+,R2
005030  012601                  MOV     (SP)+,R1

005032  000205                  RTS     R5              ;RETURN TO CALLING ROUTINE
```

MOVSB -- MOVE A STRING OF BYTES

```
                                .SBTTL  MOVSB -- MOVE A STRING OF BYTES
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ; MOVSB -- MOVE A STRING OF BYTES
                                ;
                                ;       CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,MOVSB
                                ;       .WORD   <ADDRESS OF SOURCE STRING>
                                ;       .WORD   <ADDRESS OF DESTINATION STRING>
                                ;       .WORD   <# OF BYTES TO MOVE>
                                ;
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

005034  010146          MOVSB:  MOV     R1,-(SP)        ;SAVE THE REGISTERS WE'LL BE USING
005036  010246                  MOV     R2,-(SP)
005040  010346                  MOV     R3,-(SP)

005042  012501                  MOV     (R5)+,R1        ;INITIALIZE SOURCE POINTER
005044  012502                  MOV     (R5)+,R2        ;       DESTINATION POINTER
005046  012503                  MOV     (R5)+,R3        ;       COUNTER

005050  112122          1$:     MOVB    (R1)+,(R2)+     ;MOVE IN 1 BYTE OF DATA
005052  077302                  SOB     R3,1$           ;IF MORE DATA, LOOP
                                                        ;ELSE, RESTORE REGISTERS AND RETURN

005054  012603                  MOV     (SP)+,R3        ;RESTORE REGISTERS
005056  012602                  MOV     (SP)+,R2
005060  012601                  MOV     (SP)+,R1

005062  000205                  RTS     R5              ;RETURN TO CALLING ROUTINE
```

```
                                .SBTTL  XORSW -- XOR TWO WORD TABLES
                                ;++*************************************************************************
                                ; XORSW -- DEVELOP THE EXCLUSIVE OR'S BETWEEN TWO STRINGS OF WORDS
                                ;
                                ;       CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,XORSW
                                ;       .WORD   <ADDRESS OF FIRST SOURCE STRING>
                                ;       .WORD   <ADDRESS OF SECOND SOURCE STRING>
                                ;       .WORD   <ADDRESS OF "XOR" STRING>
                                ;       .WORD   <# OF BYTES TO MOVE>
                                ;
                                ;--*************************************************************************

005064  010146                  XORSW:  MOV     R1,-(SP)        ;SAVE THE REGISTERS WE'LL BE USING
005066  010246                          MOV     R2,-(SP)
005070  010346                          MOV     R3,-(SP)
005072  010446                          MOV     R4,-(SP)

005074  012501                          MOV     (R5)+,R1        ;INITIALIZE SOURCE POINTER # 1
005076  012502                          MOV     (R5)+,R2        ;       SOURCE POINTER # 2
005100  012503                          MOV     (R5)+,R3        ;       "XOR" STRING POINTER
005102  012504                          MOV     (R5)+,R4        ;       COUNTER

005104  010546                          MOV     R5,-(SP)        ;NOW WE CAN SAVE R5 FOR THE RETURN

005106  012113                  1$:     MOV     (R1)+,(R3)      ;MOVE ONE WORD TO THE DESTINATION FIELD
005110  012205                          MOV     (R2)+,R5        ;GET SECOND WORD & SETUP FOR XOR INSTRUCTION
005112  074523                          XOR     R5,(R3)+        ;PERFORM ACTUAL XOR
005114  077404                          SOB     R4,1$           ;IF MORE DATA, LOOP
                                                                ;ELSE, RESTORE REGISTERS AND RETURN

005116  012605                          MOV     (SP)+,R5        ;RESTORE REGISTERS
005120  012604                          MOV     (SP)+,R4
005122  012603                          MOV     (SP)+,R3
005124  012602                          MOV     (SP)+,R2
005126  012601                          MOV     (SP)+,R1

005130  000205                          RTS     R5              ;RETURN TO CALLING ROUTINE


                                .SBTTL  STALL -- DELAY FOR 10.5 MICRO-SEC'S (ON LSI 11)
                                ;++*************************************************************************
                                ; STALL -- THIS SUBROUTINE STALLS FOR ABOUT 10.5 MICRO-SECONDS
                                ;--*************************************************************************

005132  000207                  STALL:  RTS     PC
```

```
                                .SBTTL  NPREAD -- "READ" CONTENTS OF ALL NPR REGISTERS

                                ;**************************************************************************
                                ;   NPREAD -- READ ALL NPR REGISTERS INTO LOC'S STARTING @ "W0"
                                ;**************************************************************************

005134  004537  003616       NPREAD: JSR     R5,READ
005140  123004                       NPRCTL
005142  002740                       BT2
005144  103447                       BCS     10$             ;ON ERROR, EXIT
005146  004537  003616               JSR     R5,READ
005152  123001                       NPRDRH
005154  002742                       BT2+2
005156  103442                       BCS     10$             ;ON ERROR, EXIT
005160  004537  003616               JSR     R5,READ
005164  123000                       NPRDRL
005166  002744                       BT2+4
005170  103435                       BCS     10$             ;ON ERROR, EXIT
005172  004537  003616               JSR     R5,READ
005176  000072                       NPRAOX
005200  002746                       BT2+6
005202  103430                       BCS     10$             ;ON ERROR, EXIT
005204  004537  003616               JSR     R5,READ
005210  000071                       NPRAOH
005212  002750                       BT2+8.
005214  103423                       BCS     10$             ;ON ERROR, EXIT
005216  004537  003616               JSR     R5,READ
005222  000070                       NPRAOL
005224  002752                       BT2+10.
005226  103416                       BCS     10$             ;ON ERROR, EXIT
005230  004537  003616               JSR     R5,READ
005234  000076                       NPRAIX
005236  002754                       BT2+12.
005240  103411                       BCS     10$             ;ON ERROR, EXIT
005242  004537  003616               JSR     R5,READ
005246  000075                       NPRAIH
005250  002756                       BT2+14.
005252  103404                       BCS     10$             ;ON ERROR, EXIT
005254  004537  003616               JSR     R5,READ
005260  000074                       NPRAIL
005262  002760                       BT2+16.

005264  000207              10$:     RTS     PC              ;RETURN
```

J5

NPRMOV -- WORD/BYTE BLOCK MOVE USING THE NPR HARDWARE

```
                                .SBTTL  NPRMOV -- WORD/BYTE BLOCK MOVE USING THE NPR HARDWARE
                                ;**************************************************************************
                                ; NPRMOV -- MOVE A BLOCK OF DATA THROUGH THE DMV'S NPR LOGIC
                                ;--************************************************************************

                                ; - - - - - - - - - I N I T I A L I Z A T I O N - - - - - - - - - - - - - - - -

005266  010146          NPRMOV: MOV     R1,-(SP)        ;SAVE THE REGISTERS WE USE
005270  010246                  MOV     R2,-(SP)
005272  010346                  MOV     R3,-(SP)

005274  012501          1$:     MOV     (R5)+,R1        ;POINT TO TEST PATTERN
005276  012502                  MOV     (R5)+,R2        ;POINT TO THE LSI-11 BUFFER AREA
005300  012503                  MOV     (R5)+,R3        ;GET COUNT OF # OF WORDS IN TEST PATTERN
005302  012537  005612          MOV     (R5)+,42$       ;LOAD UP THE COMMAND TO BE USED

005306  005037  002276          CLR     ERRFLG          ;INITIALIZE ERROR FLAG

005312  032737  000040  005612  BIT     #NPRIO,42$      ;DETERMINE DIRECTION:
005320  001403                  BEQ     3$              ;LSI ***> DMV   --  USE "NPRAIL"
005322  012746  000070          MOV     #NPRAOL,-(SP)   ;DMV ***> LSI   --  USE "NPRAOL"
005326  000402                  BR      4$
005330  012746  000074  3$:     MOV     #NPRAIL,-(SP)
005334  011637  005400  4$:     MOV     (SP),10$        ;SETUP LOW BYTE ADDRESS POINTER
005340  005216                  INC     (SP)            ;INCREMENT TO NEXT DMV ADDRESS
005342  011637  005422          MOV     (SP),14$        ;SETUP HIGH BYTE ADDRESS POINTER
005346  005216                  INC     (SP)            ;INCREMENT TO NEXT DMV ADDRESS
005350  012637  005434          MOV     (SP)+,17$       ;SETUP EXT. BYTE ADDRESS POINTER & RESTORE SP

                                ; - - - - - - - - - RE-INITIALIZE LOOP'S VARIABLES - - - - - - - - - - - - - - -

005354                  6$:

                                ; - - - - - - - - - SETUP APPROPRIATE ADDRESS - - - - - - - - - - - - - - - - -

005354  032737  000040  005612  BIT     #NPRIO,42$      ;DIRECTION OF TRANSFER?
005362  001002                  BNE     8$              ;DMV ***> LSI -- ADDR. SHOULD POINT TO BUFFER
005364  010246                  MOV     R2,-(SP)        ;LSI ***> DMV -- SAVE BUFFER POINTER
005366  010102                  MOV     R1,R2           ;       & POINT TO TEST DATA TO BE READ
005370  010237  005402  8$:     MOV     R2,11$          ;SETUP LOW BYTE OF ADDRESS
005374  004537  004054          JSR     R5,WRITEI       ;LOAD UP DESTINATION ADDRESS
005400  000070          10$:    NPRAOL                  ;  NPR ADDRESS LOW BYTE
005402  000000          11$:    0                       ;*** MODIFIED FROM ABOVE ***
005404  103510                  BCS     45$             ;ON ERROR, EXIT

005406  000302                  SWAB    R2              ;SETUP HIGH BYTE OF DESTINATION ADDRESS
005410  010237  005424          MOV     R2,15$
005414  000302                  SWAB    R2              ;  (RESTORE ADDRESS)
005416  004537  004054          JSR     R5,WRITEI       ;  SEND IT TO THE DMV
005422  000071          14$:    NPRAOH                  ;  NPR ADDRESS HIGH BYTE
005424  000000          15$:    0                       ;*** MODIFIED FROM ABOVE ***
005426  103477                  BCS     45$             ;ON ERROR, EXIT

005430  004537  004054          JSR     R5,WRITEI       ;MAKE SURE "EXTENDED" BITS ARE CLEARED
005434  000072          17$:    NPRAOX                  ;  NPR ADDRESS EXTENDED BYTE
005436  000000                  0                       ;  ACTUAL VALUE LOADED INTO THIS BYTE (WE HOPE!)
005440  103472                  BCS     45$             ;ON ERROR, EXIT
```

NPRMOV -- WORD/BYTE BLOCK MOVE USING THE NPR HARDWARE

```
2736
2737                                    ;  - - - - - - - - S E T U P   D A T A - - - - - - - - - - - - - - - - - - - - - -
2738
2739 005442  112137  005556                    MOVB    (R1)+,30$       ;SETUP ONE BYTE OF DATA TO BE PASSED TO THE DMV
2740 005446  032737  000040  005612            BIT     @NPRIO,42$      ;DIRECTION OF TRANSFER?
2741 005454  001010                            BNE     19$             ;DMV ***> LSI -- "DATA-OUT"
2742                                                                    ;LSI ***> DMV -- "DATA-IN"
2743
2744                                    ; ON AN "LSI ***> DMV", R2 WAS STACKED & IT SHOULD NOW BE RESTORED:
2745
2746 005456  012602                            MOV     (SP)+,R2        ;RESTORE R2
2747
2748                                    ; WHEN WE FALL THROUGH TO HERE, THE DIRECTION IS LSI ***> DMV -- ("DATA-IN").
2749                                    ; THIS MUST ALWAYS BE A WORD TRANSFER SO THAT BOTH HIGH & LOW BYTES MUST BE
2750                                    ; LOADED WITH THE BACKGROUND PATTERN (THE ONE'S COMPLEMENT OF THE TEST PATTERN)
2751
2752 005460  005137  005556                    COM     30$             ;ONE'S COMPLEMENT THE LOW BYTE
2753 005464  112137  005600                    MOVB    (R1)+,35$       ;SETUP THE HIGH BYTE AND
2754 005470  005137  005600                    COM     35$             ;  COMPLEMENT IT ALSO
2755 005474  000425                            BR      28$             ;NO GO AROUND THE "DATA-OUT" SETUP AND PASS
2756                                                                    ;THE BACKGROUND PATTERN TO THE DMV'S NPR DATA REG'S
2757
2758                                    ; IF WE GET TO HERE, THE DIRECTION IS DMV ***> LSI -- ("DATA-OUT").
2759                                    ; IF THIS IS A WORD XFER, BOTH HIGH & LOW BYTES MUST BE SETUP; IF BYTE, THEN
2760                                    ; WE MUST DETERMINE WHICH ONE (HIGH OR LOW) IS TO BE DONE AND ONLY LOAD THAT
2761                                    ; BYTE.  CONTRARY TO NORMAL PDP-11 OPERATION FOR A BYTE OPERATION FROM A
2762                                    ; REGISTER, THE ODD ADDRESSED BYTE WILL ALWAYS BE WRITTEN FROM THE HIGH DATA
2763                                    ; BYTE REGISTER AND THE EVEN ADDRESSED BYTE WILL ALWAYS BE WRITTEN FROM THE
2764                                    ; LOW DATA BYTE REGISTER.
2765
2766
2767 005476  113712  005556    19$:    MOVB    30$,(R2)        ;SETUP THE BACKGROUND PATTERN IN LSI'S MEM.
2768 005502  105112                    COMB    (R2)            ;  (THIS MAKES IT A "BACKGROUND" PATTERN)
2769
2770 005504  032737  000010  005612            BIT     @NPRBYT,42$     ;IS THIS A BYTE OR WORD TRANSFER?
2771 005512  001007                            BNE     21$             ;BYTE, GO DETERMINE WHICH ONE
2772 005514  111137  005600                    MOVB    (R1),35$        ;WORD, SETUP THE HIGH BYTE OF NPR DATA
2773 005520  112162  000001                    MOVB    (R1)+,1(R2)     ;ALSO, SETUP THE HIGH BYTE'S BACKGROUND PATTERN
2774 005524  105162  000001                    COMB    1(R2)           ;  (THIS MAKES IT A "BACKGROUND" PATTERN)
2775 005530  000407                            BR      28$             ;GO LOAD UP THE NPR DATA REG'S NOW
2776
2777 005532  032702  000001    21$:    BIT     #1,R2           ;IS LSI ADDRESS ODD OR EVEN (HIGH OR LOW BYTE)?
2778 005536  001404                    BEQ     28$             ;EVEN (LOW BYTE)  EVERYTHING'S OK -- GO LOAD IT
2779 005540  013737  005556  005600    MOV     30$,35$         ;ODD (HIGH BYTE).  WE SETUP THE WRONG ONE --
2780                                                            ;       PUT THE DATA BYTE IN THE RIGHT PLACE
2781 005546  000411                    BR      33$             ;  AND GO LOAD IT INTO THE DMV'S NPR HIGH DATA BYTE REG.
2782
2783
2784 005550  004537  004054    28$:    JSR     R5,WRITEI       ;LOAD UP THE
2785 005554  123000                    NPRDRL                  ;  NPR DATA REG. LOW BYTE
2786 005556  000000    30$:    0               ;  DATA BYTE -- LOW
2787 005560  103422                    BCS     45$             ;ON ERROR, EXIT
2788
2789 005562  032737  000010  005612            BIT     @NPRBYT,42$     ;IS THIS A BYTE OR WORD TRANSFER?
2790 005570  001005                            BNE     40$             ;BYTE, THEN LEAVE THE NEXT BYTE OF DATA FOR LATER
2791
2792 005572  004537  004054    33$:    JSR     R5,WRITEI       ;WORD, LOAD UP THE
```

```
2793 005576  123001                        NPRDRH                  ;  NPR DATA REG. HIGH BYTE WITH
2794 005600  000000              35$:      0                       ;  WHAT WE HOPE IS THE APPROPRIATE DATA!
2795 005602  103411                        BCS     45$             ;ON ERROR, EXIT
2796
2797                             ; - - - - - - - - - INITIATE ONE TRANSFER - - - - - - - - - - - - - - - - - - -
2798
2799
2800 005604  004537  004054      40$:      JSR     R5,WRITEI       ;MAKE THE "NPR" LOGIC DO IT'S THING
2801 005610  123004                        NPRCTL                  ;  BY LOADING THE CONTROL REGISTER WITH
2802 005612  000044              42$:      NPRDL                   ;  THE PASSED COMMAND
2803 005614  103404                        BCS     45$             ;ON ERROR, EXIT
2804
2805                             ; - - - - - - - - - MAKE SURE THE XFER RAN OK - - - - - - - - - - - - - - - - -
2806
2807 005616  004537  003730                JSR     R5,READI        ;GET THE CONTROL REG. FOR IT'S STATUS
2808 005622  123004                        NPRCTL
2809 005624  000000              44$:      0
2810 005626  103513              45$:      BCS     63$             ;ON ERROR, EXIT
2811
2812                             ;  BY NOW, THE TRANSFER SHOULD BE COMPLETE SO BIT 6 SHOULD BE = 0.  THEREFOR
2813                             ;  THE ONLY SIGNIFICIANT BIT IS BIT 7 WHICH SHOULD = 0 IF EVERYTHING WENT OK.
2814
2815 005630  132737  000200  005624        BITB    @NPRABT,44$     ;DID THE TRANSFER ABORT?
2816 005636  001436                        BEQ     50$             ;NO,  PROCEED WITH TESTING
2817 005640  005237  002412                INC     TMP0            ;YES, COUNT THE TIME-OUT
2818 005644  023727  002412  000001        CMP     TMP0,#1         ;IS THIS THE FIRST OCCURANCE OF A TIMEOUT?
2819 005652  001030                        BNE     50$             ;NO,  THEN DON'T STACK THE ERROR MESSAGE
2820                                                              ;YES, QUEUE UP A FATAL ERROR MESSAGE
2821                                                              ;PASS TO ERROR HANDLER:
2822 005654  013737  005612  002414        MOV     42$,TMP1        ;       CONTROL REGISTER AS WE SET IT
2823 005662  013737  005624  002416        MOV     44$,TMP2        ;       CONTROL REGISTER AS READ
2824 005670  113737  005424  002421        MOVB    15$,TMP3+1      ;       LSI'S MEMORY ADDRESS
2825 005676  113737  005402  002420        MOVB    11$,TMP3
2826 005704                                GTDF    EM26E,ERR11     ;IN ORDER TO REPORT THE TIME-OUT ERROR,
                                                                  ;       QUEUE "DEVICE FATAL" ERROR # 6
     005704  012737  000001  002202                                                    MOV     #T.EDF,ERRTYP
     005712  012737  000006  002204                                                    MOV     #6,ERRNBR
     005720  012737  016316  002206                                                    MOV     #EM26E,ERRMSG
     005726  012737  007314  002210                                                    MOV     #ERR11,ERRBLK
2827         000006              NPRTOE = $E                      ;THIS GETS THE ERROR # FOR TESTING LATER
2828
2829                             ; - - - - - - - - - IF "LSI ===> DMV", RETRIEVE DATA - - - - - - - - - - - - - -
2830
2831 005734  032737  000040  005612  50$:      BIT     #NPRIO,42$      ;DIRECTION?
2832 005742  001025                        BNE     54$             ;DMV ===> LSI -- DATA ALREADY IN LSI-11
2833                                                              ;LSI ===> DMV -- DATA IN REG'S,  RETRIEVE IT
2834
2835 005744  010237  005756                MOV     R2,51$          ;POINT TO LSI'S INPUT BUFFER
2836 005750  004537  003616                JSR     R5,READ         ;GET ONE BYTE
2837 005754  123000                        NPRDRL                  ;    FROM THE LOW ORDER HALF OF THE DATA REG.
2838 005756  000000              51$:      0                       ;*** MODIFIED FROM ABOVE *** DESTINATION ADDR.
2839 005760  103436                        BCS     63$             ;ON ERROR, EXIT
2840 005762  005202                        INC     R2              ;POINT TO NEXT BYTE OF THE BUFFER
2841
2842 005764  032737  000010  005612        BIT     #NPRBYT,42$     ;WAS A BYTE OF WORD "NPR" PERFORMED?
2843 005772  001022                        BNE     56$             ;BYTE  - EVERYTHING'S "KOOL"!
2844                                                              ;WORD -- MOVE HIGH BYTE INTO BUFFER
```

M5

NPRMOV -- WORD/BYTE BLOCK MOVE USING THE NPR HARDWARE

```
2845 005774  010237  006006                MOV     R2,52$          ;POINT TO LSI'S I/P BUFFER
2846 006000  004537  003616                JSR     R5,READ         ;GET ONE BYTE
2847 006004  123001                        NPRDRH                  ;    FROM THE HIGH ORDER HALF OF THE DATA REG.
2848 006006  000000              52$:      0                       ;*** MODIFIED FROM ABOVE *** DESTINATION ADDR.
2849 006010  103422                        BCS     63$             ;ON ERROR, EXIT
2850 006012  005202                        INC     R2              ;POINT TO NEXT BYTE OF THE BUFFER
2851 006014  000411                        BR      56$             ;DONE RETRIEVING DATA -- CHECK FOR MORE
2852
2853                           ;- - - - - - - - - DMV ***> LSI -- JUST ADVANCE LSI-11 ADDRESS - - - - - - - -
2854
2855 006016  005202              54$:      INC     R2              ;BUMP THE LSI-11 ADDRESS
2856 006020  032737  000010  005612        BIT     #NPRBYT,42$     ;IS THIS A BYTE OR WORD TRANSFER?
2857 006026  001004                        BNE     56$             ;BYTE, THEN ADDRESS IS OK AS IS
2858 006030  005202                        INC     R2              ;WORD, BUMP ADDR. -- WE ALREADY DID THE HIGH BYTE
2859 006032  000402                        BR      56$
2860
2861                           ;- - - - - - - - - T E S T   F O R   M O R E - - - - - - - - - - - - - - - - -
2862
2863 006034  000137  005354              55$:      JMP     6$              ;THIS LITTLE BIT IF CUTE LOGIC IS NECESSARY
2864                                                                      ;BECAUSE "6$" IS TOO FAR AWAY FOR A BRANCH!
2865 006040  077303              56$:      SOB     R3,55$          ;DO IT AGAIN IF THERE IS MORE DATA
2866 006042  005737  002276                TST     ERRFLG          ;WAS AN ERROR DETECTED?
2867 006046  001402                        BEQ     61$             ;NO,  TAKE NORMAL EXIT
2868
2869                           ;- - - - - - - - - C L E A N   U P   &   E X I T - - - - - - - - - - - - - - - -
2870
2871 006050  000261              60$:      SEC                     ;INDICATE ERROR CONDITION
2872 006052  000401                        BR      63$
2873
2874 006054  000241              61$:      CLC                     ;INDICATE NO ERROR
2875
2876 006056  012603              63$:      MOV     (SP)+,R3        ;RESTORE THE REGISTERS AGAIN
2877 006060  012602                        MOV     (SP)+,R2
2878 006062  012601                        MOV     (SP)+,R1
2879
2880 006064  000205                        RTS     R5              ;RETURN
```

```
                                .SBTTL  INTERRUPT HANDLER -- MPIHAN

                                ;+*****************************************************************************
                                ; MPIHAN -- COUNT INTERRUPTS -- USUALLY INTERRUPT "A"
                                ;
                                ;       THIS ROUTINE WILL INCREMENT THE LOW BYTE OF "INTFLG" EACH TIME IT IS
                                ;       ENTERED.  IF "IHILNK" IS NON-ZERO, VECTOR TO THE ADDRESS THEREIN USING
                                ;       A "JSR  PC"
                                ;-*****************************************************************************

006066                          BGNSRV  MPIHAN
006066                                                                  MPIHAN::
006066  010046                  MOV     R0,-(SP)        ;SAVE R0
006070  105737  002274          TSTB    INTWCH          ;HAVE WE BEEN TOLD TO WATCH FOR TYPE "A" INT'S?
006074  001007                  BNE     5$              ;YES, DO NORMAL INTERRUPT PROCESSING
006076  004737  004232          JSR     PC,GETBSR       ;NO,  DUMP REGISTERS AND
006102                          GEDF    EM34,ERR3       ;       REPORT "UNEXPECTED INTERRUPT"
                                                        ;       "DEVICE FATAL" ERROR # 7
006102  104455                                                          TRAP    C$ERDF
006104  000007                                                          .WORD   7
006106  016526                                                          .WORD   EM34
006110  006220                                                          .WORD   ERR3
006112  000407                  BR      10$             ;GO TO EXIT

006114  105237  002272          5$:     INCB    INTFLG          ;INCREMENT LOW BYTE OF INTERRUPT COUNTER
006120  005737  006136                  TST     IHILNK          ;ARE WE EXPECTED TO EXECUTE ANOTHER ROUTINE?
006124  001402                  BEQ     10$             ;NO, GET OUT
006126  004777  000004          JSR     PC,@IHILNK      ;YES, GO TO IT -- I HOPE IT'S VALID!
006132  012600                  10$:    MOV     (SP)+,R0        ;RESTORE R0
006134                          ENDSRV                  ;RETURN TO INTERRUPTED PROCESS
006134                                                                  L10002:
006134  000002                                                                  RTI

006136  000000                  IHILNK: .WORD   0               ;POINTER TO AUXILIARY INT. HANDLING ROUTINE
```

INTERRUPT HANDLER MPOHAN

```
                        .SBTTL  INTERRUPT HANDLER -- MPOHAN
                        ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                        ; MPOHAN -- SIMPLY COUNT INTERRUPTS -- USUALLY INTERRUPT "B"
                        ;
                        ;       THIS ROUTINE WILL INCREMENT THE HIGH BYTE OF "INTFLG" EACH TIME IT IS
                        ;       ENTERED.  IF "IHOLNK" IS NON-ZERO, VECTOR TO THE ADDRESS THEREIN USING
                        ;       A "JSR  PC"
                        ;
                        ;---------------------------------------------------------------------------------------

006140                          BGNSRV  MPOHAN
006140                                                                  MPOHAN::
006140  010046                  MOV     R0,-(SP)        ;SAVE R0
006142  105737  002275          TSTB    INTWCH+1        ;HAVE WE BEEN TOLD TO WATCH FOR TYPE "B" INT'S?
006146  001007                  BNE     5$              ;YES, DO NORMAL INTERRUPT PROCESSING
006150  004737  004232          JSR     PC,GETBSR       ;NO,  DUMP REGISTERS AND
006154                          GEDF    EM34B,ERR3      ;       REPORT "UNEXPECTED INTERRUPT"
                                                        ;       "DEVICE FATAL" ERROR # 8
006154  104455                                                          TRAP    C$ERDF
006156  000010                                                          .WORD   8
006160  016540                                                          .WORD   EM34B
006162  006220                                                          .WORD   ERR3
006164  000407                  BR      10$             ;GO TO EXIT

006166  105237  002273  5$:     INCB    INTFLG+1        ;INCREMENT HIGH BYTE OF INTERRUPT COUNTER
006172  005737  006210          TST     IHOLNK          ;ARE WE EXPECTED TO EXECUTE ANOTHER ROUTINE?
006176  001402                  BEQ     10$             ;NO, GET OUT
006200  004777  000004          JSR     PC,@IHOLNK      ;YES, GO TO IT -  I HOPE IT'S VALID!
006204  012600          10$:    MOV     (SP)+,R0        ;RESTORE R0
006206                          ENDSRV                  ;RETURN TO INTERRUPTED PROCESS
006206                                                                  L10003:
006206  000002                                                                  RTI

006210  000000  IHOLNK:  .WORD   0               ;POINTER TO AUXILIARY INT. HANDLING ROUTINE
```

C6

GLOBAL ERROR REPORT REPORT SECTION

```
                        .SBTTL  GLOBAL ERROR REPORT REPORT SECTION

                        ;//////////////////////////////////////////////////////////////////////////////
                        ;/      THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
                        ;/      THAT ARE USED IN MORE THAN ONE TEST.
                        ;//////////////////////////////////////////////////////////////////////////////
                            .EVEN
                        ;------------------------------------------------------------------------------
                        .SBTTL  ERROR HANDLER -- ERR1 -- "NO NOTHING" HANDLER
                        ;------------------------------------------------------------------------------
006212                      BGNMSG  ERR1
006212                                                              ERR1::
006212  004737  013076      JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
006216                      ENDMSG
006216                                                              L10004:
006216  104423                                                          TRAP    C$MSG
                        ;------------------------------------------------------------------------------
                        .SBTTL  ERROR HANDLER -- ERR3 -- DUMP THE BYTE SELECT REGISTERS
                        ;------------------------------------------------------------------------------
006220                      BGNMSG  ERR3
006220                                                              ERR3::
006220  004737  012206      JSR     PC,ERR4$
006224  004737  013076      JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
006230                      ENDMSG
006230                                                              L10005:
006230  104423                                                          TRAP    C$MSG
                        ;------------------------------------------------------------------------------
                        .SBTTL  ERROR HANDLER -- ERR4 -- M-LOOP TIMEOUT ERROR HANDLING
                        ;------------------------------------------------------------------------------
006232                      BGNMSG  ERR4
006232                                                              ERR4::
006232  010146          MOV     R1,-(SP)            ;SAVE THE WORKING REGISTER
006234  113701  002254      MOVB    GDATA,R1        ;SAVE THIS FOR LATER
006240  122701  000017      CMPB    #17,R1          ;WAS THIS AN M-LOOP REQUEST?
006244  103013              BHIS    5$              ;YES, THEN REPORT THE FUNCTION CODE
006246                      PRINTX  #FMT5,<B,R1>    ;NO.  THEN IT MUST BE A BSEL1 SETTING
006246  005046                                                          CLR     -(SP)
006250  150116                                                          BISB    R1,(SP)
006252  012746  013346                                                  MOV     #FMT5,-(SP)
006256  012746  000002                                                  MOV     #2,-(SP)
006262  010600                                                          MOV     SP,R0
006264  104415                                                          TRAP    C$PNTX
006266  062706  000006                                                  ADD     #6,SP
006272  000425          BR      20$
006274  001001      5$: BNE     6$                  ;IF IT WAS A 17, THIS IS A "NOP" AND
006276  005001          CLR     R1                  ;  THE TEXT POINTER MUST SO REFLECT.
006300  022701  000007  6$: CMP     #7,R1           ;IS FUNCTION CODE > 7?
006304  002002          BGE     7$                  ;NO.  THAN WE CAN HANDLE IT
006306  012701  000006      MOV     #6,R1           ;YES. THAN IT'S UNDEFINED - SAY SO
006312  006301          7$: ASL     R1              ;CONVERT TO A WORD OFFSET
006314                      PRINTX  #FMT5A,<B,GDATA>,TXTMLT(R1)     ;REPORT THE FAILING FUNCTION
006314  016146  021746                                                  MOV     TXTMLT(R1),-(SP)
006320  005046                                                          CLR     -(SP)
006322  153716  002254                                                  BISB    GDATA,(SP)
006326  012746  013411                                                  MOV     #FMT5A,-(SP)
006332  012746  000003                                                  MOV     #3,-(SP)
```

```
ERROR HANDLER -- ERR4 -- M-LOOP TIMEOUT ERROR HANDLING

        006336  010600                                                          MOV     SP,R0
        006340  104415                                                          TRAP    C$PNTX
        006342  062706  000010                                                  ADD     #10,SP
 2974
 2975 006346  012601                    20$:     MOV     (SP)+,R1            ;RESTORE THE WORKING REGISTER
 2976 006350                             PRINTX  #FMT4,#TXT6,#TXT4
        006350  012746  015212                                                  MOV     #TXT4,-(SP)
        006354  012746  015313                                                  MOV     #TXT6,-(SP)
        006360  012746  013213                                                  MOV     #FMT4,-(SP)
        006364  012746  000003                                                  MOV     #3,-(SP)
        006370  010600                                                          MOV     SP,R0
        006372  104415                                                          TRAP    C$PNTX
        006374  062706  000010                                                  ADD     #10,SP
 2977 006400                             PRINTX  #FMT11,WSR0,WSR2,WSR4,WSR6 ;DUMP THE SELECT REGISTERS
        006400  013746  002220                                                  MOV     WSR6,-(SP)
        006404  013746  002216                                                  MOV     WSR4,-(SP)
        006410  013746  002214                                                  MOV     WSR2,-(SP)
        006414  013746  002212                                                  MOV     WSR0,-(SP)
        006420  012746  013770                                                  MOV     #FMT11,-(SP)
        006424  012746  000005                                                  MOV     #5,-(SP)
        006430  010600                                                          MOV     SP,R0
        006432  104415                                                          TRAP    C$PNTX
        006434  062706  000014                                                  ADD     #14,SP
 2978 006440                             PRINTX  #FMT4B,#TXT4A
        006440  012746  015252                                                  MOV     #TXT4A,-(SP)
        006444  012746  013306                                                  MOV     #FMT4B,-(SP)
        006450  012746  000002                                                  MOV     #2,-(SP)
        006454  010600                                                          MOV     SP,R0
        006456  104415                                                          TRAP    C$PNTX
        006460  062706  000006                                                  ADD     #6,SP
 2979 006464                             PRINTX  #FMT11,WSR10,WSR12,WSR14,WSR16
        006464  013746  002230                                                  MOV     WSR16,-(SP)
        006470  013746  002226                                                  MOV     WSR14,-(SP)
        006474  013746  002224                                                  MOV     WSR12,-(SP)
        006500  013746  002222                                                  MOV     WSR10,-(SP)
        006504  012746  013770                                                  MOV     #FMT11,-(SP)
        006510  012746  000005                                                  MOV     #5,(SP)
        006514  010600                                                          MOV     SP,R0
        006516  104415                                                          TRAP    C$PNTX
        006520  062706  000014                                                  ADD     #14,SP
 2980 006524  004737  013076            JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
 2981 006530                             ENDMSG
        006530                                                          L10006:
        006530  104423                                                          TRAP    C$MSG
 2982
 2983                           ;-------------------------------------------------------------------------
                                .SBTTL  ERROR HANDLER -- ERR8 -- NPR REGISTER ERRORS
 2984                           ;-------------------------------------------------------------------------
 2985 006532                            BGNMSG  ERR8
        006532                                                          ERR8::
 2986 006532  113701  002300            MOVB    REGNUM,R1       ;THIS WAS CALCULATED TO BE A WORD OFFSET
 2987 006536                            PRINTB  #FMT07,#TXTNP,TXTNPT(R1)
        006536  016146  021770                                                  MOV     TXTNPT(R1),-(SP)
        006542  012746  015745                                                  MOV     #TXTNP,-(SP)
        006546  012746  013476                                                  MOV     #FMT07,-(SP)
        006552  012746  000003                                                  MOV     #3,-(SP)
        006556  010600                                                          MOV     SP,R0
        006560  104414                                                          TRAP    C$PNTB
```

ERROR HANDLER -- ERR8 -- NPR REGISTER ERRORS

```
      006562  062706  000010                                                    ADD     #10,SP
 2988 006566                          PRINTX  #FMT06A
      006566  012746  013542                                                    MOV     #FMT06A,-(SP)
      006572  012746  000001                                                    MOV     #1,-(SP)
      006576  010600                                                            MOV     SP,R0
      006600  104415                                                            TRAP    C$PNTx
      006602  062706  000004                                                    ADD     #4,SP
 2989
 2990                                 ;       PRINT FIRST SET OF REGISTERS: CONTROL & DATA
 2991
 2992 006606                          PRINTX  #FMT06,#TXT11A
      006606  012746  015437                                                    MOV     #TXT11A,-(SP)
      006612  012746  013530                                                    MOV     #FMT06,-(SP)
      006616  012746  000002                                                    MOV     #2,-(SP)
      006622  010600                                                            MOV     SP,R0
      006624  104415                                                            TRAP    C$PNTx
      006626  062706  000006                                                    ADD     #6,SP
 2993 006632  010146                  MOV     R1,-(SP)        ;PRESERVE R1
 2994 006634  013701  002254          MOV     GDATA,R1        ;POINTER TO EXPECTED DATA
 2995 006640                          PRINTX  #FMT16,#TXT8A,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
      006640  005046                                                            CLR     -(SP)
      006642  152116                                                            BISB    (R1)+,(SP)
      006644  005046                                                            CLR     -(SP)
      006646  152116                                                            BISB    (R1)+,(SP)
      006650  005046                                                            CLR     -(SP)
      006652  152116                                                            BISB    (R1)+,(SP)
      006654  012746  015336                                                    MOV     #TXT8A,-(SP)
      006660  012746  014007                                                    MOV     #FMT16,-(SP)
      006664  012746  000005                                                    MOV     #5,-(SP)
      006670  010600                                                            MOV     SP,R0
      006672  104415                                                            TRAP    C$PNTx
      006674  062706  000014                                                    ADD     #14,SP
 2996 006700  013701  002256          MOV     BDATA,R1        ;POINTER TO ACTUAL DATA
 2997 006704                          PRINTX  #FMT16,#TXT8B,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
      006704  005046                                                            CLR     -(SP)
      006706  152116                                                            BISB    (R1)+,(SP)
      006710  005046                                                            CLR     -(SP)
      006712  152116                                                            BISB    (R1)+,(SP)
      006714  005046                                                            CLR     -(SP)
      006716  152116                                                            BISB    (R1)+,(SP)
      006720  012746  015353                                                    MOV     #TXT8B,-(SP)
      006724  012746  014007                                                    MOV     #FMT16,-(SP)
      006730  012746  000005                                                    MOV     #5,-(SP)
      006734  010600                                                            MOV     SP,R0
      006736  104415                                                            TRAP    C$PNTx
      006740  062706  000014                                                    ADD     #14,SP
 2998 006744  004537  005064          JSR     R5,XORSW        ;GENERATE XOR'S
 2999 006750  002254                  .WORD   GDATA           ;BETWEEN GOOD DATA
 3000 006752  002256                  .WORD   BDATA           ;AND BAD DATA
 3001 006754  003054                  .WORD   W0              ;AND PUT THEM HERE
 3002 006756  000011                  .WORD   9.              ;ONLY DO THIS MANY
 3003 006760                          PRINTX  #FMT16,#TXT8C,<B,W0>,<B,W1>,<B,W2>
      006760  005046                                                            CLR     -(SP)
      006762  153716  003060                                                    BISB    W2,(SP)
      006766  005046                                                            CLR     -(SP)
      006770  153716  003056                                                    BISB    W1,(SP)
      006774  005046                                                            CLR     -(SP)
```

ERROR HANDLER -- ERR8 -- NPR REGISTER ERRORS

```
        006776  153716  003054                                                          BISB    W0,(SP)
        007002  012746  015370                                                          MOV     #TXT8C,-(SP)
        007006  012746  014007                                                          MOV     #FMT16,-(SP)
        007012  012746  000005                                                          MOV     #5,-(SP)
        007016  010600                                                                  MOV     SP,R0
        007020  104415                                                                  TRAP    C$PNTX
        007022  062706  000014                                                          ADD     #14,SP
  3004
  3005                                  ;       PRINT SECOND SET OF REGISTERS: ADDRESS' OUT & IN
  3006
  3007 007026                           PRINTX  #FMT06,#TXT11B
        007026  012746  015460                                                          MOV     #TXT11B,-(SP)
        007032  012746  013530                                                          MOV     #FMT06,-(SP)
        007036  012746  000002                                                          MOV     #2,-(SP)
        007042  010600                                                                  MOV     SP,R0
        007044  104415                                                                  TRAP    C$PNTX
        007046  062706  000006                                                          ADD     #6,SP
  3008 007052  012701  002257           MOV     #GDATA+3,R1     ;POINTER TO EXPECTED DATA
  3009 007056                           PRINTX  #FMT16A,#TXT8A,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
        007056  005046                                                                 CLR     -(SP)
        007060  152116                                                                 BISB    (R1)+,(SP)
        007062  005046                                                                 CLR     -(SP)
        007064  152116                                                                 BISB    (R1)+,(SP)
        007066  005046                                                                 CLR     -(SP)
        007070  152116                                                                 BISB    (R1)+,(SP)
        007072  005046                                                                 CLR     -(SP)
        007074  152116                                                                 BISB    (R1)+,(SP)
        007076  005046                                                                 CLR     -(SP)
        007100  152116                                                                 BISB    (R1)+,(SP)
        007102  005046                                                                 CLR     -(SP)
        007104  152116                                                                 BISB    (R1)+,(SP)
        007106  012746  015336                                                          MOV     #TXT8A,-(SP)
        007112  012746  014032                                                          MOV     #FMT16A,-(SP)
        007116  012746  000010                                                          MOV     #10,-(SP)
        007122  010600                                                                  MOV     SP,R0
        007124  104415                                                                  TRAP    C$PNTX
        007126  062706  000022                                                          ADD     #22,SP
  3010 007132  012701  002261           MOV     #BDATA+3,R1     ;POINTER TO ACTUAL DATA
  3011 007136                           PRINTX  #FMT16A,#TXT8B,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
        007136  005046                                                                 CLR     -(SP)
        007140  152116                                                                 BISB    (R1)+,(SP)
        007142  005046                                                                 CLR     -(SP)
        007144  152116                                                                 BISB    (R1)+,(SP)
        007146  005046                                                                 CLR     -(SP)
        007150  152116                                                                 BISB    (R1)+,(SP)
        007152  005046                                                                 CLR     -(SP)
        007154  152116                                                                 BISB    (R1)+,(SP)
        007156  005046                                                                 CLR     -(SP)
        007160  152116                                                                 BISB    (R1)+,(SP)
        007162  005046                                                                 CLR     -(SP)
        007164  152116                                                                 BISB    (R1)+,(SP)
        007166  012746  015353                                                          MOV     #TXT8B,-(SP)
        007172  012746  014032                                                          MOV     #FMT16A,-(SP)
        007176  012746  000010                                                          MOV     #10,-(SP)
        007202  010600                                                                  MOV     SP,R0
        007204  104415                                                                  TRAP    C$PNTX
        007206  062706  000022                                                          ADD     #22,SP
```

ERROR HANDLER -- ERR8 -- NPR REGISTER ERRORS

```
3012 007212  012701  003062                MOV     #W3,R1          ;POINT TO REST OF XOR'S
3013 007216                                PRINTX  #FMT16A,#TXT8C,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>,<B,(R1)+>
     007216  005046                                                                        CLR     -(SP)
     007220  152116                                                                        BISB    (R1)+,(SP)
     007222  005046                                                                        CLR     -(SP)
     007224  152116                                                                        BISB    (R1)+,(SP)
     007226  005046                                                                        CLR     -(SP)
     007230  152116                                                                        BISB    (R1)+,(SP)
     007232  005046                                                                        CLR     -(SP)
     007234  152116                                                                        BISB    (R1)+,(SP)
     007236  005046                                                                        CLR     -(SP)
     007240  152116                                                                        BISB    (R1)+,(SP)
     007242  005046                                                                        CLR     -(SP)
     007244  152116                                                                        BISB    (R1)+,(SP)
     007246  012746  015370                                                              MOV     #TXT8C,-(SP)
     007252  012746  014032                                                              MOV     #FMT16A,-(SP)
     007256  012746  000010                                                              MOV     #10.,-(SP)
     007262  010600                                                                      MOV     SP,R0
     007264  104415                                                                      TRAP    C$PNTX
     007266  062706  000022                                                              ADD     #22,SP
3014 007272  004737  013076                JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3015 007276                                ENDMSG
     007276                                                                         L10007:
     007276  104423                                                                        TRAP    C$MSG
3016                                ;-------------------------------------------------------------------------------
3017                                .SBTTL  ERROR HANDLER -- ERR9 -- WORD NPR I/O ERRORS
3018                                ;-------------------------------------------------------------------------------
3019 007300                                BGNMSG  ERR9
     007300                                                                         ERR9::
3020 007300  004737  012614                JSR     PC,ERR9.        ;USE COMMON "ERROR 9" ROUTINE
3021 007304                                ENDMSG
     007304                                                                         L10010:
     007304  104423                                                                        TRAP    C$MSG
3022                                ;-------------------------------------------------------------------------------
3023                                .SBTTL  ERROR HANDLER -- ERR10 -- BYTE NPR I/O ERRORS
3024                                ;-------------------------------------------------------------------------------
3025 007306                                BGNMSG  ERR10
     007306                                                                         ERR10::
3026 007306  004737  012752                JSR     PC,ERR10.       ;USE COMMON "ERROR 10" ROUTINE
3027 007312                                ENDMSG
     007312                                                                         L10011:
     007312  104423                                                                        TRAP    C$MSG
3028                                ;-------------------------------------------------------------------------------
3029                                .SBTTL  ERROR HANDLER -- ERR11 -- NPR TIMEOUT ERRORS
3030                                ;-------------------------------------------------------------------------------
3031 007314                                BGNMSG  ERR11
     007314                                                                         ERR11::
3032 007314  023727  002412  000001        CMP     TMPO,#1         ;IF ONLY ONE TIMEOUT,
3033 007322  001412                        BEQ     1$              ;NO NEED TO TELL HOW MANY OCCURED
3034 007324                                PRINTX  #FMT17C,TMPO    ;ELSE,  SAY HOW MANY WERE FOUND IN ALL
     007324  013746  002412                                                              MOV     TMPO,-(SP)
     007330  012746  014307                                                              MOV     #FMT17C,(SP)
     007334  012746  000002                                                              MOV     #2,-(SP)
     007340  010600                                                                      MOV     SP,R0
     007342  104415                                                                      TRAP    C$PNTX
     007344  062706  000006                                                              ADD     #6,SP
3035 007350                                1$:     PRINTX  #FMT17,<B,TMP1> ;NPRCTL SENT
```

ERROR HANDLER -- ERR11 -- NPR TIMEOUT ERRORS

```
        007350  005046                                                          CLR     -(SP)
        007352  153716  002414                                                  BISB    TMP1,(SP)
        007356  012746  014074                                                  MOV     #FMT17,-(SP)
        007362  012746  000002                                                  MOV     #2,-(SP)
        007366  010600                                                          MOV     SP,R0
        007370  104415                                                          TRAP    C$PNTX
        007372  062706  000006                                                  ADD     #6,SP
3036    007376                  PRINTX  #FMT17A,<B,TMP2>  ;NPRCTL       READ
        007376  005046                                                          CLR     -(SP)
        007400  153716  002416                                                  BISB    TMP2,(SP)
        007404  012746  014153                                                  MOV     #FMT17A,-(SP)
        007410  012746  000002                                                  MOV     #2,-(SP)
        007414  010600                                                          MOV     SP,R0
        007416  104415                                                          TRAP    C$PNTX
        007420  062706  000006                                                  ADD     #6,SP
3037    007424                  PRINTX  #FMT17B,TMP3      ;LSI-11'S MEMORY ADDRESS
        007424  013746  002420                                                  MOV     TMP3,-(SP)
        007430  012746  014234                                                  MOV     #FMT17B,-(SP)
        007434  012746  000002                                                  MOV     #2,-(SP)
        007440  010600                                                          MOV     SP,R0
        007442  104415                                                          TRAP    C$PNTX
        007444  062706  000006                                                  ADD     #6,SP
3038    007450  004737  013076          JSR     PC,NULERR        ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3039    007454                  ENDMSG
        007454                                                          L10012:
        007454  104423                                                         TRAP    C$MSG
3040            ;-----------------------------------------------------------------------------
3041            .SBTTL  ERROR HANDLER -- ERR12 -- NPR EXTENDED ADDRESSING ERROR HANDLER
3042            ;-----------------------------------------------------------------------------
3043    007456          BGNMSG  ERR12
        007456                                                          ERR12::
3044    007456          PRINTX  #FMT12           ;PRINT FIRST HEADING LINE
        007456  012746  007672                                                  MOV     #FMT12,-(SP)
        007462  012746  000001                                                  MOV     #1,-(SP)
        007466  010600                                                          MOV     SP,R0
        007470  104415                                                          TRAP    C$PNTX
        007472  062706  000004                                                  ADD     #4,SP
3045    007476          PRINTX  #FMT12A          ;PRINT SECOND HEADING LINE
        007476  012746  007732                                                  MOV     #FMT12A,-(SP)
        007502  012746  000001                                                  MOV     #1,-(SP)
        007506  010600                                                          MOV     SP,R0
        007510  104415                                                          TRAP    C$PNTX
        007512  062706  000004                                                  ADD     #4,SP
3046                                     ;PRINT ADDRESS, CONTROL, & EXPECTED DATA
3047    007516  004737  012162          JSR     PC,xORGB         ;GENERATE EXCLUSIVE OR OF EXPECTED & READ DATA
3048    007522          PRINTX  #FMT12D,<B,TMPF+1>,#0,#0,<B,TMP2>,GDATA,BDATA,XDATA
        007522  013746  002260                                                  MOV     XDATA,(SP)
        007526  013746  002256                                                  MOV     BDATA,-(SP)
        007532  013746  002254                                                  MOV     GDATA,-(SP)
        007536  005046                                                          CLR     -(SP)
        007540  153716  002416                                                  BISB    TMP2,(SP)
        007544  012746  000000                                                  MOV     #0,-(SP)
        007550  012746  000000                                                  MOV     #0,-(SP)
        007554  005046                                                          CLR     -(SP)
        007556  153716  002451                                                  BISB    TMPF+1,(SP)
        007562  012746  010020                                                  MOV     #FMT12D,-(SP)
        007566  012746  000010                                                  MOV     #10,(SP)
```

ERROR HANDLER -- ERR12 -- NPR EXTENDED ADDRESSING ERROR HANDLE

```
        007572  010600                                                          MOV     SP,R0
        007574  104415                                                          TRAP    C$PNTX
        007576  062706  000022                                                  ADD     #22,SP
3049    007602  023737  002256  002420          CMP     BDATA,TMP3      ;DID WE READ THE BACKGROUND PATTERN?
3050    007610  001011                          BNE     4$              ;NO,
3051    007612                                  PRINTX  #FMT12E         ;YES, INDICATE WRONG PAGE READ
        007612  012746  010063                                                  MOV     #FMT12E,-(SP)
        007616  012746  000001                                                  MOV     #1,-(SP)
        007622  010600                                                          MOV     SP,R0
        007624  104415                                                          TRAP    C$PNTX
        007626  062706  000004                                                  ADD     #4,SP
3052    007632  000414                          BR      60$             ;       AND EXIT ERROR HANDLER
3053
3054    007634  023737  002256  002414  4$:     CMP     BDATA,TMP1      ;DID WE EVEN PERFORM A READ?
3055    007642  001010                          BNE     60$             ;YES, THEN WE CAN GIVE ANY FURTHER ERROR INFO.
3056    007644                                  PRINTX  #FMT12F         ;NO,  THEN WE CAN AT LEAST SAY THAT MUCH
        007644  012746  010145                                                  MOV     #FMT12F,-(SP)
        007650  012746  000001                                                  MOV     #1,-(SP)
        007654  010600                                                          MOV     SP,R0
        007656  104415                                                          TRAP    C$PNTX
        007660  062706  000004                                                  ADD     #4,SP
3057    007664  004737  013076          60$:    JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3058    007670                                  ENDMSG
        007670                                                          L10013:
        007670  104423                                                          TRAP    C$MSG
3059                                    .NLIST  BEX
3060    007672      045     116     045 FMT12:  .ASCIZ  /%N%S13%ANPR REGISTERS%S14%ADATA/
3061    007732      045     116     045 FMT12A: .ASCIZ  /%N%S11%AADDRESS   CONTROL   EXPECTED   READ     XOR%N/
3062    010020      045     116     045 FMT12D: .ASCIZ  /%N%S8%03%S%03%S%03%S%03%S3%03%010%09%09/
3063    010063      045     116     045 FMT12E: .ASCIZ  /%N%S7%A(NPR OPERATION ACCESSED WRONG MEMORY PAGE)/
3064    010145      045     116     045 FMT12F: .ASCIZ  /%N%S2%A(NPR DATA REGISTER UN-CHANGED FROM BEFORE REQUEST)/
3065                                    .LIST   BEX
3066                                    .EVEN
3067                                    ;-----------------------------------------------------------------------
3068                                    .SBTTL  ERROR HANDLER -- ERR13 -- "MMU" ERROR HANDLER
3069                                    ;-----------------------------------------------------------------------
3070    010240                          BGNMSG  ERR13
        010240                                                          ERR13::
3071    010240                                  PRINTX  #FMT13A
        010240  012746  010546                                                  MOV     #FMT13A,-(SP)
        010244  012746  000001                                                  MOV     #1,-(SP)
        010250  010600                                                          MOV     SP,R0
        010252  104415                                                          TRAP    C$PNTX
        010254  062706  000004                                                  ADD     #4,SP
3072    010260                                  PRINTX  #FMT13B
        010260  012746  010611                                                  MOV     #FMT13B,-(SP)
        010264  012746  000001                                                  MOV     #1,-(SP)
        010270  010600                                                          MOV     SP,R0
        010272  104415                                                          TRAP    C$PNTX
        010274  062706  000004                                                  ADD     #4,SP
3073    010300                                  PRINTX  #FMT13C
        010300  012746  010656                                                  MOV     #FMT13C,-(SP)
        010304  012746  000001                                                  MOV     #1,-(SP)
        010310  010600                                                          MOV     SP,R0
        010312  104415                                                          TRAP    C$PNTX
        010314  062706  000004                                                  ADD     #4,SP
3074    010320                                  PRINTX  #FMT13D
```

```
        010320  012746  010721                                                  MOV     #FMT13D,-(SP)
        010324  012746  000001                                                  MOV     #1,-(SP)
        010330  010600                                                          MOV     SP,R0
        010332  104415                                                          TRAP    C$PNTX
        010334  062706  000004                                                  ADD     #4,SP
  3075  010340                          PRINTX  #FMT11,TMP0,TMP1,TMP2,TMP3
        010340  013746  002420                                                  MOV     TMP3,-(SP)
        010344  013746  002416                                                  MOV     TMP2,-(SP)
        010350  013746  002414                                                  MOV     TMP1,-(SP)
        010354  013746  002412                                                  MOV     TMP0,-(SP)
        010360  012746  013770                                                  MOV     #FMT11,-(SP)
        010364  012746  000005                                                  MOV     #5,-(SP)
        010370  010600                                                          MOV     SP,R0
        010372  104415                                                          TRAP    C$PNTX
        010374  062706  000014                                                  ADD     #14,SP
  3076  010400                          PRINTX  #FMT13E,TMP4,TMP5,TMP6,TMP7
        010400  013746  002430                                                  MOV     TMP7,-(SP)
        010404  013746  002426                                                  MOV     TMP6,-(SP)
        010410  013746  002424                                                  MOV     TMP5,-(SP)
        010414  013746  002422                                                  MOV     TMP4,-(SP)
        010420  012746  010764                                                  MOV     #FMT13E,-(SP)
        010424  012746  000005                                                  MOV     #5,-(SP)
        010430  010600                                                          MOV     SP,R0
        010432  104415                                                          TRAP    C$PNTX
        010434  062706  000014                                                  ADD     #14,SP
  3077  010440                          PRINTX  #FMT11,REG0,REG1,REG2,REG3
        010440  013746  002400                                                  MOV     REG3,-(SP)
        010444  013746  002376                                                  MOV     REG2,-(SP)
        010450  013746  002374                                                  MOV     REG1,-(SP)
        010454  013746  002372                                                  MOV     REG0,-(SP)
        010460  012746  013770                                                  MOV     #FMT11,-(SP)
        010464  012746  000005                                                  MOV     #5,-(SP)
        010470  010600                                                          MOV     SP,R0
        010472  104415                                                          TRAP    C$PNTX
        010474  062706  000014                                                  ADD     #14,SP
  3078  010500                          PRINTX  #FMT13E,REG4,REG5,REG6,REG7
        010500  013746  002410                                                  MOV     REG7,-(SP)
        010504  013746  002406                                                  MOV     REG6,-(SP)
        010510  013746  002404                                                  MOV     REG5,-(SP)
        010514  013746  002402                                                  MOV     REG4,-(SP)
        010520  012746  010764                                                  MOV     #FMT13E,-(SP)
        010524  012746  000005                                                  MOV     #5,-(SP)
        010530  010600                                                          MOV     SP,R0
        010532  104415                                                          TRAP    C$PNTX
        010534  062706  000014                                                  ADD     #14,SP
  3079  010540  004737  013076                JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
  3080  010544                          ENDMSG
        010544                                                          L10014:
        010544  104423                                                          TRAP    C$MSG
  3081                          .NLIST  BEX
  3082  010546      045     116     045 FMT13A: .ASCIZ  /%N%A    SR0     SR1     SR2     SR3/
  3083  010611      045     101     040 FMT13B: .ASCIZ  /%A     KPAR6   KPDR6  V ADDR   DATA?/
  3084  010656      045     116     045 FMT13C: .ASCIZ  /%N%A     R0      R1      R2      R3/
  3085  010721      045     101     040 FMT13D: .ASCIZ  /%A      R4      R5      SP      PC/
  3086  010764      045     117     070 FMT13E: .ASCIZ  /%08%08%08%08%08/
  3087                          .LIST   BEX
  3088                          .EVEN
```

```
3089                                    ;-------------------------------------------------------------------------
3090                                    .SBTTL  ERROR HANDLER -- ERR14 -- NPR REGISTER LOAD ERROR HANDLER
3091                                    ;-------------------------------------------------------------------------
3092 011002                                    BGNMSG  ERR14
     011002                                                                                      ERR14::
3093 011002  010146                            MOV     R1,-(SP)        ;SAVE GENERAL REGISTER
3094 011004  013701  002300                    MOV     REGNUM,R1
3095 011010  006301                            ASL     R1              ;CONVERT REG # TO WORD INDEX
3096 011012                                    PRINTB  #FMT07,#TXTNP,TXTNPT(R1)
     011012  016146  021770                                                                     MOV     TXTNPT(R1),-(SP)
     011016  012746  015745                                                                     MOV     #TXTNP,-(SP)
     011022  012746  013476                                                                     MOV     #FMT07 -(SP)
     011026  012746  000003                                                                     MOV     #3,-(SP)
     011032  010600                                                                             MOV     SP,R0
     011034  104414                                                                             TRAP    C$PNTB
     011036  062706  000010                                                                     ADD     #10,SP
3097 011042  004737  012162                    JSR     PC,XORGB
3098                                                                    ;DATA: GOOD, BAD, & XOR
3099 011046  023727  002300  000006            CMP     REGNUM,#6       ;IF EXTENDED ADDRESS BYTE, USE BYTE PRINT
3100 011054  001423                            BEQ     5$
3101 011056  023727  002300  000003            CMP     REGNUM,#3
3102 011064  001417                            BEQ     5$
3103                                                                    ;ELSE, USE WORD PRINTS
3104 011066                                    PRINTX  #FMT10,GDATA,BDATA,XDATA
     011066  013746  002260                                                                     MOV     XDATA,-(SP)
     011072  013746  002256                                                                     MOV     BDATA,-(SP)
     011076  013746  002254                                                                     MOV     GDATA,-(SP)
     011102  012746  013714                                                                     MOV     #FMT10,-(SP)
     011106  012746  000004                                                                     MOV     #4,-(SP)
     011112  010600                                                                             MOV     SP,R0
     011114  104415                                                                             TRAP    C$PNTX
     011116  062706  000012                                                                     ADD     #12,SP
3105 011122  000421                            BR      10$             ;BYPASS BYTE PRINTS IF WORD PRINTS USED
3106
3107 011124                            5$:     PRINTX  #FMT02A,<B,GDATA>,<B,BDATA>,<B,XDATA>
     011124  005046                                                                             CLR     -(SP)
     011126  153716  002260                                                                     BISB    XDATA,(SP)
     011132  005046                                                                             CLR     -(SP)
     011134  153716  002256                                                                     BISB    BDATA,(SP)
     011140  005046                                                                             CLR     -(SP)
     011142  153716  002254                                                                     BISB    GDATA,(SP)
     011146  012746  013127                                                                     MOV     #FMT02A,-(SP)
     011152  012746  000004                                                                     MOV     #4,-(SP)
     011156  010600                                                                             MOV     SP,R0
     011160  104415                                                                             TRAP    C$PNTX
     011162  062706  000012                                                                     ADD     #12,SP
3108 011166  004737  013076            10$:    JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3109 011172  012601                            MOV     (SP)+,R1        ;RESTORE GENERAL REGISTER
3110 011174                                    ENDMSG
     011174                                                                                     L10015:
     011174  104423                                                                             TRAP    C$MSG
3111
3112                                    ;-------------------------------------------------------------------------
3113                                    .SBTTL  ERROR HANDLER -- ERR51 -- FOR REPORTING TIMER # 2 ERRORS
3114                                    ;-------------------------------------------------------------------------
3115 011176                                    BGNMSG  ERR51
     011176                                                                                      ERR51::
```

ERROR HANDLER -- ERR51 -- FOR REPORTING TIMER # 2 ERRORS

```
3116 011176  010146                    MOV     R1,-(SP)        ;SAVE R1 FOR CALLER
3117 011200  113701  002441            MOVB    TMP8+1,R1       ;GET THE MODE LAST SETUP
3118 011204  000241                    CLC                     ;SEEING AS THE CARRY BIT WILL BE ROTATED INTO
3119                                                           ;THE DATA, WE HAD BETTER CLEAR IT JUST IN CASE.
3120 011206  042701  177737            BIC     #^C<BIT5>,R1    ;LOOK @ JUST THE TIMER 2 MODE DEFINITION
3121 011212  106101                    ROLB    R1              ;POSITION IT FOR PRINTOUT
3122 011214  106101                    ROLB    R1
3123 011216  106101                    ROLB    R1
3124 011220  106101                    ROLB    R1
3125
3126                                   ;IDENTIFY THE MODE BEING USED AT THE TIME, AND THE VALUES THAT WERE
3127                                   ;LOADED INTO THE LATCHES:
3128
3129 011222                            PRINTX  #FMT51A,R1,<B,TMP9+1>,<B,TMP8+1>
     011222  005046                                                            CLR     -(SP)
     011224  153716  002433                                                    BISB    TMP8+1,(SP)
     011230  005046                                                            CLR     -(SP)
     011232  153716  002435                                                    BISB    TMP9+1,(SP)
     011236  010146                                                            MOV     R1,-(SP)
     011240  012746  014673                                                    MOV     #FMT51A,-(SP)
     011244  012746  000004                                                    MOV     #4,-(SP)
     011250  010600                                                            MOV     SP,R0
     011252  104415                                                            TRAP    C$PNTX
     011254  062706  000012                                                    ADD     #12,SP
3130 011260  004737  013076            JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3131 011264  012601                    MOV     (SP)+,R1        ;RESTORE R1 FOR CALLER
3132 011266                            ENDMSG
     011266                                                                 L10016:
     011266  104423                                                            TRAP    C$MSG
3133
3134                                   ;-------------------------------------------------------------------
3135                                   .SBTTL  ERROR HANDLER -- ERR52 -- PROCESS SHIFT REGISTER ERROR MESSAGES
3136 011270                            ;-------------------------------------------------------------------
     011270                            BGNMSG  ERR52
                                                                       ERR52::
3137 011270  004537  003616            JSR     R5,READ         ;GET CURRENT VALUES WITHIN ACR & PCR
3138 011274  120013                    ACR
3139 011276  002440                    TMPB
3140 011300  004537  003616            JSR     R5,READ
3141 011304  120014                    PCR
3142 011306  002442                    TMPC
3143 011310                            PRINTX  #FMT52H
     011310  012746  011514                                                    MOV     #FMT52H,-(SP)
     011314  012746  000001                                                    MOV     #1,-(SP)
     011320  010600                                                            MOV     SP,R0
     011322  104415                                                            TRAP    C$PNTX
     011324  062706  000004                                                    ADD     #4,SP
3144 011330                            PRINTX  #FMT52A,#TXT8D,<B,TMPA+1>,<B,TMPB+1>,<B,TMPE+1>
     011330  005046                                                            CLR     -(SP)
     011332  153716  002447                                                    BISB    TMPE+1,(SP)
     011336  005046                                                            CLR     -(SP)
     011340  153716  002441                                                    BISB    TMPB+1,(SP)
     011344  005046                                                            CLR     -(SP)
     011346  153716  002437                                                    BISB    TMPA+1,(SP)
     011352  012746  015405                                                    MOV     #TXT8D,-(SP)
     011356  012746  011557                                                    MOV     #FMT52A,-(SP)
     011362  012746  000005                                                    MOV     #5,-(SP)
     011366  010600                                                            MOV     SP,R0
```

```
        011370  104415                                                                  TRAP    C$PNTX
        011372  062706  000014                                                          ADD     #14,SP
3145    011376                          PRINTX  #FMT52B,#TXT8E,<B,TMPA>,<B,TMPB>,<B,TMPC>,<B,TMPD>,<B,TMPE>
        011376  005046                                                                  CLR     -(SP)
        011400  153716  002446                                                          BISB    TMPE,(SP)
        011404  005046                                                                  CLR     -(SP)
        011406  153716  002444                                                          BISB    TMPD,(SP)
        011412  005046                                                                  CLR     -(SP)
        011414  153716  002442                                                          BISB    TMPC,(SP)
        011420  005046                                                                  CLR     -(SP)
        011422  153716  002440                                                          BISB    TMPB,(SP)
        011426  005046                                                                  CLR     -(SP)
        011430  153716  002436                                                          BISB    TMPA,(SP)
        011434  012746  015422                                                          MOV     #TXT8E,-(SP)
        011440  012746  011607                                                          MOV     #FMT52B,-(SP)
        011444  012746  000007                                                          MOV     #7,-(SP)
        011450  010600                                                                  MOV     SP,R0
        011452  104415                                                                  TRAP    C$PNTX
        011454  062706  000020                                                          ADD     #20,SP
3146    011460  005737  002252                 TST     TDATA           ;HAS "SR" BEEN WRITTEN YET?
3147    011464  001010                         BNE     10$             ;NO,  THEN JUST FINISH ERROR MESSAGE
3148    011466                                 PRINTX  #FMT52C         ;YES, ADD THAT INFORMATION TO MESSAGE
        011466  012746  011652                                                          MOV     #FMT52C,-(SP)
        011472  012746  000001                                                          MOV     #1,-(SP)
        011476  010600                                                                  MOV     SP,R0
        011500  104415                                                                  TRAP    C$PNTX
        011502  062706  000004                                                          ADD     #4,SP
3149    011506  004737  013076          10$:   JSR     PC,NULERR       ;USE COMMON ROUTINE TO TERMINATE ERROR MESSAGE
3150    011512                                 ENDMSG
        011512                                                                  L10017:
        011512  104423                                                                 TRAP    C$MSG
3151
3152                                    .NLIST  BEX
3153    011514      045     116     045 FMT52H: .ASCIZ  /%N%S14%ASR   ACR   PCR   IFR   IER/
3154    011557      045     116     045 FMT52A: .ASCIZ  /%N%S1%T%03%S3%03%S15%03/
3155    011607      045     116     045 FMT52B: .ASCIZ  /%N%S1%T%03%S3%03%S3%03%S3%03%S3%03%S3%03/
3156    011652      045     116     045 FMT52C: .ASCIZ  /%N%S10%A(SR HASN'T BEEN LOADED YET!)/
3157                                    .LIST   BEX
3158                                    .EVEN
3159                                    ;       SR ERROR FORMATS:
3160                                    ;
3161                                    ;         SR    ACR   PCR   IFR   IER
3162                                    ; LOADED XXX   XXX   ---   ---   XXX
3163                                    ; READ   XXX   XXX   XXX   XXX   XXX
3164                                    ;
3165                                    ;-------------- -- ------------------------------------------------------------
3166
3167                                    ;-------------------------------------------------------------------------
3168                                    .SBTTL  ERROR HANDLER -- ERR60 -- NPR WRITE-EXTENDED BIT ERRORS
3169                                    ;-------------------------------------------------------------------------
3170    011720                                 BGNMSG  ERR60
        011720                                                                  ERR60::
3171    011720                                 PRINTX  #FMT60
        011720  012746  012034                                                          MOV     #FMT60,-(SP)
        011724  012746  000001                                                          MOV     #1,-(SP)
        011730  010600                                                                  MOV     SP,R0
        011732  104415                                                                  TRAP    C$PNTX
```

```
     011734  062706  000004                                                    ADD     #4,SP
3172 011740                          PRINTX  #FMT61
     011740  012746  012073                                                    MOV     #FMT61,-(SP)
     011744  012746  000001                                                    MOV     #1,-(SP)
     011750  010600                                                            MOV     SP,R0
     011752  104415                                                            TRAP    C$PNTX
     011754  062706  000004                                                    ADD     #4,SP
3173 011760  005003                          CLR     R3                 ;CLEAR INDEX
3174 011762                    10$:  PRINTX  #FMT62,XLOC0(R3),XVAL0(R3),RXVAL0(R3)
     011762  016346  032516                                                    MOV     RXVAL0(R3),-(SP)
     011766  016346  032500                                                    MOV     XVAL0(R3),-(SP)
     011772  016346  032462                                                    MOV     XLOC0(R3),-(SP)
     011776  012746  012134                                                    MOV     #FMT62,-(SP)
     012002  012746  000004                                                    MOV     #4,-(SP)
     012006  010600                                                            MOV     SP,R0
     012010  104415                                                            TRAP    C$PNTX
     012012  062706  000012                                                    ADD     #12,SP
3175 012016  005723                          TST     (R3)+              ;BUMP INDEX
3176 012020  020327  000016                  CMP     R3,#14.
3177 012024  001356                          BNE     10$
3178 012026  004737  013076                  JSR     PC,NULERR
3179 012032                          ENDMSG
     012032                                                          L10020:
     012032  104423                                                            TRAP    C$MSG
3180                           .NLIST  BEX
3181 012034    045    116    045  FMT60:  .ASCIZ /%N%S3%AXLOC%S6%AXVAL%S6%ARXVAL/
3182 012073    045    116    045  FMT61:  .ASCIZ /%N%S3%A----%S6%A----%S6%A-----%N/
3183 012134    045    116    045  FMT62:  .ASCIZ /%N%S2%06%S4%06%S4%06/
3184                           .LIST   BEX
3185                           .EVEN
```

ERROR HANDLER SUBROUTINES

```
                                .SBTTL  ERROR HANDLER SUBROUTINES
                                ;
                                ;       SUBROUTINES USED ONLY BY ERROR HANDLERS
                                ;
                                .SBTTL          ERROR HANDLER SUBROUTINE  .ORGB
                                ;
                                ;       PERFORM EXCLUSIVE OR BETWEEN "GDATA" & "BDATA" PUTTING
                                ;       THE RESULT IN "XDATA"
012162  010146                  XORGB:  MOV     R1,-(SP)        ;PRESERVE WORKING REGISTER
012164  013701  002254                  MOV     GDATA,R1        ;GET "GOOD" DATA
012170  013737  002256  002260          MOV     BDATA,XDATA     ;AND "BAD" DATA
012176  074137  002260                  XOR     R1,XDATA        ;PERFORM EXCLUSIVE OR
012202  012601                          MOV     (SP)+,R1        ;RESTORE R1
012204  000207                          RTS     PC              ;RETURN
                                ;
                                .SBTTL          ERROR HANDLER SUBROUTINE  ERR4$
                                ;
                                ;       IDENTIFY & DUMP THE BYTE SELECT REGISTERS
                                ;
012206                  ERR4$:  PRINTX  #FMT4,#TXT3,#TXT1
012206  012746  014761                  MOV     #TXT1,-(SP)
012212  012746  015162                  MOV     #TXT3,-(SP)
012216  012746  013213                  MOV     #FMT4,-(SP)
012222  012746  000003                  MOV     #3,-(SP)
012226  010600                          MOV     SP,R0
012230  104415                          TRAP    C$PNTX
012232  062706  000010                  ADD     #10,SP
012236                          PRINTX  #FMT4A,<B,BSR0>,<B,BSR1>,<B,BSR2>,<B,BSR3>
012236  005046                          CLR     -(SP)
012240  153716  002220                  BISB    BSR3,(SP)
012244  005046                          CLR     -(SP)
012246  153716  002216                  BISB    BSR2,(SP)
012252  005046                          CLR     -(SP)
012254  153716  002214                  BISB    BSR1,(SP)
012260  005046                          CLR     -(SP)
012262  153716  002212                  BISB    BSR0,(SP)
012266  012746  013253                  MOV     #FMT4A,-(SP)
012272  012746  000005                  MOV     #5,-(SP)
012276  010600                          MOV     SP,R0
012300  104415                          TRAP    C$PNTX
012302  062706  000014                  ADD     #14,SP
012306                          PRINTX  #FMT4B,#TXT2
012306  012746  015017                  MOV     #TXT2,-(SP)
012312  012746  013306                  MOV     #FMT4B,-(SP)
012316  012746  000002                  MOV     #2,-(SP)
012322  010600                          MOV     SP,R0
012324  104415                          TRAP    C$PNTX
012326  062706  000006                  ADD     #6,SP
012332                          PRINTX  #FMT4C,<B,BSR4>,<B,BSR5>,<B,BSR6>,<B,BSR7>
012332  005046                          CLR     -(SP)
012334  153716  002230                  BISB    BSR7,(SP)
012340  005046                          CLR     -(SP)
012342  153716  002226                  BISB    BSR6,(SP)
```

ERROR HANDLER SUBROUTINE -- ERR4$

```
        012346  005046                                                          CLR     -(SP)
        012350  153716  002224                                                  BISB    BSR5,(SP)
        012354  005046                                                          CLR     -(SP)
        012356  153716  002222                                                  BISB    BSR4,(SP)
        012362  012746  013513                                                  MOV     #FMT4C,-(SP)
        012366  012746  000005                                                  MOV     #5,-(SP)
        012372  010600                                                          MOV     SP,R0
        012374  104415                                                          TRAP    C$PNTX
        012376  062706  000014                                                  ADD     #14,SP
3214    012402                          PRINTX  #FMT4B,#TXT2A
        012402  012746  015061                                                  MOV     #TXT2A,-(SP)
        012406  012746  013306                                                  MOV     #FMT4B,-(SP)
        012412  012746  000002                                                  MOV     #2,-(SP)
        012416  010600                                                          MOV     SP,R0
        012420  104415                                                          TRAP    C$PNTX
        012422  062706  000006                                                  ADD     #6,SP
3215    012426                          PRINTX  #FMT4A,<B,BSR10>,<B,BSR11>,<B,BSR12>,<B,BSR13>
        012426  005046                                                          CLR     -(SP)
        012430  153716  002240                                                  BISB    BSR13,(SP)
        012434  005046                                                          CLR     -(SP)
        012436  153716  002236                                                  BISB    BSR12,(SP)
        012442  005046                                                          CLR     -(SP)
        012444  153716  002234                                                  BISB    BSR11,(SP)
        012450  005046                                                          CLR     -(SP)
        012452  153716  002232                                                  BISB    BSR10,(SP)
        012456  012746  013253                                                  MOV     #FMT4A,-(SP)
        012462  012746  000005                                                  MOV     #5,-(SP)
        012466  010600                                                          MOV     SP,R0
        012470  104415                                                          TRAP    C$PNTX
        012472  062706  000014                                                  ADD     #14,SP
3216    012476                          PRINTX  #FMT4B,#TXT2B
        012476  012746  015120                                                  MOV     #TXT2B,-(SP)
        012502  012746  013306                                                  MOV     #FMT4B,-(SP)
        012506  012746  000002                                                  MOV     #2,-(SP)
        012512  010600                                                          MOV     SP,R0
        012514  104415                                                          TRAP    C$PNTX
        012516  062706  000006                                                  ADD     #6,SP
3217    012522                          PRINTX  #FMT4C,<B,BSR14>,<B,BSR15>,<B,BSR16>,<B,BSR17>
        012522  005046                                                          CLR     -(SP)
        012524  153716  002250                                                  BISB    BSR17,(SP)
        012530  005046                                                          CLR     -(SP)
        012532  153716  002246                                                  BISB    BSR16,(SP)
        012536  005046                                                          CLR     -(SP)
        012540  153716  002244                                                  BISB    BSR15,(SP)
        012544  005046                                                          CLR     -(SP)
        012546  153716  002242                                                  BISB    BSR14,(SP)
        012552  012746  013313                                                  MOV     #FMT4C,(SP)
        012556  012746  000005                                                  MOV     #5,-(SP)
        012562  010600                                                          MOV     SP,R0
        012564  104415                                                          TRAP    C$PNTX
        012566  062706  000014                                                  ADD     #14,SP
3218    012572  000207                 RTS     PC
3219
3220    ;.....
3221    ;.SBTTL         ERROR HANDLER SUBROUTINE -- ERR9$ & ERR9,
3222    ;.....
3223    ;       COMMON ERROR 9 ROUTINE TO IDENTIFY THE FAILING ADDRESS & DATA
```

ERROR HANDLER SUBROUTINE -- ERR9$ & ERR9.

```
3224
3225 012574                        ERR9$:  PRINTX  #NEWLIN          ;WHEN CALLED FROM TEST, START A NEW LINE
     012574  012746  013124                                                        MOV     #NEWLIN,-(SP)
     012600  012746  000001                                                        MOV     #1,-(SP)
     012604  010600                                                                MOV     SP,R0
     012606  104415                                                                TRAP    C$PNTX
     012610  062706  000004                                                        ADD     #4,SP
3226 012614  005237  002300        ERR9.:  INC     REGNUM           ;CONVERT INDEX TO ELEMENT #
3227 012620                                PRINTX  #FMT09,REGNUM    ;IDENTIFY DATA PATTERN OFFSET
     012620  013746  002300                                                        MOV     REGNUM,-(SP)
     012624  012746  013621                                                        MOV     #FMT09,-(SP)
     012630  012746  000002                                                        MOV     #2,-(SP)
     012634  010600                                                                MOV     SP,R0
     012636  104415                                                                TRAP    C$PNTX
     012640  062706  000006                                                        ADD     #6,SP
3228 012644  004737  012162                JSR     PC,XORGB
3229 012650                                PRINTX  #FMT10,GDATA,BDATA,XDATA  ;DATA: GOOD, BAD, & XOR
     012650  013746  002260                                                        MOV     XDATA,-(SP)
     012654  013746  002256                                                        MOV     BDATA,-(SP)
     012660  013746  002254                                                        MOV     GDATA,-(SP)
     012664  012746  013714                                                        MOV     #FMT10,-(SP)
     012670  012746  000004                                                        MOV     #4,-(SP)
     012674  010600                                                                MOV     SP,R0
     012676  104415                                                                TRAP    C$PNTX
     012700  062706  000012                                                        ADD     #12,SP
3230 012704                                PRINTX  #FMT09A,TDATA    ;LSI ADDRESS
     012704  013746  002252                                                        MOV     TDATA,-(SP)
     012710  012746  013673                                                        MOV     #FMT09A,-(SP)
     012714  012746  000002                                                        MOV     #2,-(SP)
     012720  010600                                                                MOV     SP,R0
     012722  104415                                                                TRAP    C$PNTX
     012724  062706  000006                                                        ADD     #6,SP
3231 012730  000207                        RTS     PC
3232
3233                               ;-----------------------------------------------------------------------
3234                               .SBTTL          ERROR HANDLER SUBROUTINE -- ERR10$ & ERR10.
3235                               ;-----------------------------------------------------------------------
3236                               ;       COMMON ERROR 10 ROUTINE TO IDENTIFY THE FAILING ADDRESS & DATA
3237
3238 012732                        ERR10$: PRINTX  #NEWLIN          ;WHEN CALLED FROM TEST, START A NEW LINE
     012732  012746  013124                                                        MOV     #NEWLIN,-(SP)
     012736  012746  000001                                                        MOV     #1,-(SP)
     012742  010600                                                                MOV     SP,R0
     012744  104415                                                                TRAP    C$PNTX
     012746  062706  000004                                                        ADD     #4,SP
3239 012752  005237  002300        ERR10.: INC     REGNUM           ;CONVERT INDEX TO ELEMENT #
3240 012756                                PRINTX  #FMT09,REGNUM    ;IDENTIFY DATA PATTERN OFFSET
     012756  013746  002300                                                        MOV     REGNUM,-(SP)
     012762  012746  013621                                                        MOV     #FMT09,-(SP)
     012766  012746  000002                                                        MOV     #2,-(SP)
     012772  010600                                                                MOV     SP,R0
     012774  104415                                                                TRAP    C$PNTX
     012776  062706  000006                                                        ADD     #6,SP
3241 013002  004737  012162                JSR     PC,XORGB
3242 013006                                PRINTX  #FMT02A,<B,GDATA>,<B,BDATA>,<B,XDATA>  ;DATA: GOOD, BAD, & XOR
     013006  005046                                                                CLR     -(SP)
     013010  153716  002260                                                        BISB    XDATA,(SP)
```

ERROR HANDLER SUBROUTINE -- ERR10$ & ERR10.

```
        013014  005046                                                    CLR     -(SP)
        013016  153716  002256                                            BISB    BDATA,(SP)
        013022  005046                                                    CLR     -(SP)
        013024  153716  002254                                            BISB    GDATA,(SP)
        013030  012746  013127                                            MOV     #FMT02A,-(SP)
        013034  012746  000004                                            MOV     #4,-(SP)
        013040  010600                                                    MOV     SP,R0
        013042  104415                                                    TRAP    C$PNTX
        013044  062706  000012                                            ADD     #12,SP
3243    013050                          PRINTX  #FMT09A,TDATA   ;LSI ADDRESS
        013050  013746  002252                                            MOV     TDATA,-(SP)
        013054  012746  013673                                            MOV     #FMT09A,-(SP)
        013060  012746  000002                                            MOV     #2,-(SP)
        013064  010600                                                    MOV     SP,R0
        013066  104415                                                    TRAP    C$PNTX
        013070  062706  000006                                            ADD     #6,SP
3244    013074  000207                 RTS     PC
3245
3246                            ;-------------------------------------------------------------------------------
3247                            .SBTTL          SUBROUTINE TO PERFORM "PRINTB   #ENDEMB"
3248                            ;-------------------------------------------------------------------------------
3249
3250    013076                  NULERR: PRINTB  #ENDEMB         ;TERMINATE ERROR MESSAGE
        013076  012746  013120                                            MOV     #ENDEMB,-(SP)
        013102  012746  000001                                            MOV     #1,-(SP)
        013106  010600                                                    MOV     SP,R0
        013110  104414                                                    TRAP    C$PNTB
        013112  062706  000004                                            ADD     #4,SP
3251    013116  000207                  RTS     PC
3252                            ;-------------------------------------------------------------------------------
```

FORMAT SPEC'S FOR ERROR HANDLERS -- "FMT___"

```
3254                                    .SBTTL  FORMAT SPEC'S FOR ERROR HANDLERS -- "FMT___"
3255                                    ;-----------------------------------------------------------------------
3256                                    ;------------- FORMAT SPEC'S USED BY ERROR HANDLERS ---------------------
3257                                    ;-----------------------------------------------------------------------
3258                                    .NLIST  BEX
3259 013120    045     116     062     ENDEMB: .ASCIZ  /%N2/
3260 013124    045     116     000     NEWLIN: .ASCIZ  /%N/
3261
3262 013127    045     116     045     FMT02A: .ASCIZ  /%N%A  EXPECTED:  %03%A   ACTUAL:  %03%A   XOR:  %03/
3263 013213    045     116     045     FMT4:   .ASCIZ  /%N%A  THE CONTENTS OF ALL%T%N%T/
3264 013253    045     116     045     FMT4A:  .ASCIZ  /%N%S1%03%S5%03%S5%03%S5%03/
3265 013306    045     116     045     FMT4B:  .ASCIZ  /%N%T/
3266 013313    045     116     045     FMT4C:  .ASCIZ  /%N%S5%03%S5%03%S5%03%S5%03/
3267 013346    045     116     045     FMT5:   .ASCIZ  /%N%A  WHEN %03%A LOADED INTO BSEL1/
3268 013411    045     116     045     FMT5A:  .ASCIZ  /%N%A  ATTEMPTING "M-LOOP" FUNCTION CODE %02%A (%T%A)/
3269 013476    045     101     040     FMT07:  .ASCIZ  /%A  DETECTED IN %T%T%A --/
3270 013530    045     116     062     FMT06:  .ASCIZ  /%N2%S13%T/
3271 013542    045     116     062     FMT06A: .ASCIZ  /%N2%A  N  P  R     R  E  G  I  S  T  E  R  S:/
3272 013621    045     101     040     FMT09:  .ASCIZ  /%A  DETECTED @ TEST PATTERN ELEMENT # %D2/
3273 013673    045     101     040     FMT09A: .ASCIZ  /%A  LSI ADDR:%08/
3274
3275 013714    045     116     045     FMT10:  .ASCIZ  /%N%A  EXPECTED:%08%A  ACTUAL:%08%A  XOR:%08/
3276 013770    045     116     045     FMT11:  .ASCIZ  /%N%08%08%08%08/
3277 014007    045     116     045     FMT16:  .ASCIZ  /%N%T%03%S4%03%S%03/
3278 014032    045     116     045     FMT16A: .ASCIZ  /%N%T%03%S%03%S%03%S4%03%S%03%S%03/
3279 014074    045     101     040     FMT17:  .ASCIZ  /%A    VALUE SENT TO NPR CONTROL REGISTER:  %03/
3280 014153    045     116     045     FMT17A: .ASCIZ  /%N%A    VALUE READ FROM CONTROL REGISTER:    %03/
3281 014234    045     116     045     FMT17B: .ASCIZ  /%N%A    LSI-11 MEMORY ADDRESS ACCESSED:%08/
3282 014307    045     116     045     FMT17C: .ASCIZ  /%N%A  INFORMATION ON THE FIRST OF %D5%A ERRORS:/
3283 014367    045     116     045     FMT50A: .ASCIZ  /%N%A  TIMER # 1 MODE:  %01%A   REGISTERS:/
3284 014441    045     116     045     FMT50B: .ASCIZ  /%N%S15%AT1CH  T1CL  T1LH  T1LL   ACR   IFR   IER/
3285 014522    045     116     045     FMT50C: .ASCIZ  /%N%S3%T%S1%03%S3%03%S3%03%S3%03/
3286 014562    045     123     063     FMT50D: .ASCIZ  /%S3%03%S9%03/
3287 014577    045     123     063     FMT50E: .ASCIZ  /%S3%03%S3%03/
3288 014614    045     116     062     FMT50M: .ASCIZ  /%N2%S10%A(T1CH & T1CL HAVEN'T YET BEEN LOADED)/
3289 014673    045     116     045     FMT51A: .ASCIZ  /%N%A  TIMER # 2 MODE:  %01%A   T2CH & T2LL:  %03%S%03/
3290
3291                                    .SBTTL  TEXT STRINGS FOR ERROR HANDLERS -- "TXT___"
3292                                    ;-----------------------------------------------------------------------
3293                                    ;------------- TEXT USED BY ERROR HANDLERS ------------------------------
3294                                    ;-----------------------------------------------------------------------
3295
3296 014761    102     123     105     TXT1:   .ASCIZ  /BSEL0   BSEL1   BSEL2   BSEL3/
3297 015017    040     040     040     TXT2:   .ASCIZ  /    BSEL4   BSEL5   BSEL6   BSEL7/
3298 015061    102     123     105     TXT2A:  .ASCIZ  /BSEL10  BSEL11  BSEL12  BSEL13/
3299 015120    040     040     040     TXT2B:  .ASCIZ  /   BSEL14  BSEL15  BSEL16  BSEL17/
3300 015162    040     102     131     TXT3:   .ASCIZ  / BYTE SELECT REG'S ARE:/
3301 015212    040     040     040     TXT4:   .ASCIZ  /   SEL0    SEL2    SEL4    SEL6/
3302 015252    040     040     040     TXT4A:  .ASCIZ  /   SEL10   SEL12   SEL14   SEL16/
3303 015313    040     123     105     TXT6:   .ASCIZ  / SELECT REG'S ARE:/
3304 015336    040     105     130     TXT8A:  .ASCIZ  / EXPECTED:  /
3305 015353    040     101     103     TXT8B:  .ASCIZ  / ACTUAL:    /
3306 015370    040     130     117     TXT8C:  .ASCIZ  / XOR:       /
3307 015405    040     114     117     TXT8D:  .ASCIZ  / LOADED:    /
3308 015422    040     122     105     TXT8E:  .ASCIZ  / READ:      /
3309
3310 015437    103     117     116     TXT11A: .ASCIZ  /CONTROL  D A T A/
```

TEXT STRINGS FOR ERROR HANDLERS -- "TXT..."

```
3311 015460  040  040  040  TXT11B: .ASCIZ  /   OUT ADDR.      IN ADDR./
3312 015513  021  000       TXTNUL: .BYTE   21,0                    ;CTL-Q -- THIS (WE HOPE) IS HARMLESS
3313
3314 015515  116  117  120  TXTML0: .ASCIZ  /NOP/
3315 015521  122  105  101  TXTML1: .ASCIZ  /READ 1 BYTE/
3316 015535  127  122  111  TXTML2: .ASCIZ  /WRITE 1 BYTE/
3317 015552  116  120  122  TXTML3: .ASCIZ  /NPR-OUT 256 BYTES/
3318 015574  116  120  122  TXTML4: .ASCIZ  /NPR-IN 256 BYTES/
3319 015615  123  105  124  TXTML5: .ASCIZ  /SET MICROPROCESSOR'S PC/
3320 015645  125  116  104  TXTML6: .ASCIZ  /UNDEFINED/
3321 015657  015  012  123  TXTML7: .ASCIZ  <15><12>/SET MAINT INTERRUPT & CLR INT DISABLE IN CPU STATUS/
3322
3323 015745  116  120  122  TXTNP:  .ASCIZ  /NPR /
3324 015752  103  117  116  TXTNP0: .ASCIZ  /CONTROL/
3325 015762  104  101  124  TXTNP1: .ASCIZ  /DATA HI/
3326 015772  104  101  124  TXTNP2: .ASCIZ  /DATA LO/
3327 016002  101  104  104  TXTNP3: .ASCIZ  /ADDR. OUT EX/
3328 016017  101  104  104  TXTNP4: .ASCIZ  /ADDR. OUT HI/
3329 016034  101  104  104  TXTNP5: .ASCIZ  /ADDR. OUT LO/
3330 016051  101  104  104  TXTNP6: .ASCIZ  /ADDR. IN EX/
3331 016065  101  104  104  TXTNP7: .ASCIZ  /ADDR. IN HI/
3332 016101  101  104  104  TXTNP8: .ASCIZ  /ADDR. IN LO/
```

```
                                .SBTTL  ERROR MESSAGES -- "EM___"
                                ;-------------------------------------------------------------------------
                                ;-------------- ERROR MESSAGES USED BY ERROR CALL'S ----------------------
                                ;-------------------------------------------------------------------------

016115  125  055  104  EM3:    .ASCIZ  /U-DIAG. FAILURE/
016135  115  122  104  EM4:    .ASCIZ  /MRDY TIMEOUT/
016152  116  120  122  EM26:   .ASCIZ  /NPR LOGIC M-CLEAR FAILURE/
016204  102  101  104  EM26A:  .ASCIZ  /BAD NPR REG. LOAD/
016226  127  117  122  EM26B:  .ASCIZ  /WORD NPR-OUT ERROR/
016251  102  131  124  EM26C:  .ASCIZ  /BYTE NPR-OUT ERROR/
016274  127  117  122  EM26D:  .ASCIZ  /WORD NPR-IN ERROR/
016316  116  120  122  EM26E:  .ASCIZ  /NPR TIMEOUT -- "ABORT" SET/
016351  116  120  122  EM26F:  .ASCIZ  \NPR BS7 FAILURE ON WRITE\
016402  116  120  122  EM26G:  .ASCIZ  /NPR-ABORT FAILURE/
016424  115  115  125  EM27:   .ASCIZ  /MMU ABORT!/
016437  130  055  101  EM27A:  .ASCIZ  /X-ADDR. NPR ABORT/
016461  130  055  101  EM27B:  .ASCIZ  /X-ADDR. NPR HUNG/
016502  130  055  101  EM27C:  .ASCIZ  /X-ADDR. NPR FAILURE/
016526  042  101  042  EM34:   .ASCIZ  /"A" INT.?/
016540  042  102  042  EM34B:  .ASCIZ  /"B" INT.?/
016552  115  111  123  EM35:   .ASCIZ  /MISSING "A" INT./
016573  115  111  123  EM35B:  .ASCIZ  /MISSING "B" INT./
016614  104  115  126  EM40:   .ASCIZ  /DMV INIT'D BY "BINIT" WITH "DISABL INIT" SET/
016671  042  115  101  EM41:   .ASCIZ  /"MASTER RESET" FAILED WHEN "DISABL INIT" SET/
016746  116  117  040  EM42A:  .ASCIZ  /NO "POWER UP" VECTOR ON "DCOK" GOING HIGH/
017020  116  117  040  EM42B:  .ASCIZ  /NO INIT ON "DCOK" LOW & "DISABL INIT" CLEAR/
017074  111  116  126  EM42C:  .ASCIZ  /INVALID INIT ON "DCOK" LOW & "DISABL INIT" SET/
017153  104  115  126  EM43A:  .ASCIZ  \DMV MICRO-CODE HUNG\
017177  042  110  101  EM43B:  .ASCIZ  /"HALT" FAILED/
017215  116  117  040  EM43C:  .ASCIZ  /NO POWER-UP SEQUENCE/
017242  042  115  117  EM43D:  .ASCIZ  /"MOP-BOOT" LOAD FAILED/
017271  042  124  061  EM50A:  .ASCIZ  \"T1" FLAG NOT CLEARED BY LOADING T1LH\
017337  042  124  061  EM50B:  .ASCIZ  \"T1" FLAG NOT CLEARED BY LOADING T1CH\
017405  042  124  061  EM50C:  .ASCIZ  \"T1" FLAG NOT CLEARED BY READING T1CL\
017453  126  111  101  EM50D:  .ASCIZ  \VIA'S T1CL NOT DECREMENTING\
017507  126  111  101  EM50E:  .ASCIZ  \VIA'S T1CH NOT DECREMENTING\
017543  042  124  061  EM50F:  .ASCIZ  \"T1" FLAG NOT SET ON TIMER 1 TIMEOUT\
017610  042  124  061  EM50G:  .ASCIZ  \"T1" FLAG CLEARED BY READING T1CH\
017652  126  111  101  EM50H:  .ASCIZ  \VIA'S T1LL IMPROPERLY LOADED BY WRITING T1CL @ ADDR 4\
017740  042  124  061  EM50I:  .ASCIZ  \"T1" FLAG CLEARED BY READING T1LL\
020002  126  111  101  EM50J:  .ASCIZ  \VIA'S T1LH IMPROPERLY LOADED BY WRITING T1CH @ ADDR 5\
020070  042  124  061  EM50K:  .ASCIZ  \"T1" FLAG CLEARED BY READING T1LH\
020132  042  124  061  EM50L:  .ASCIZ  \"T1" FLAG NOT SET AFTER RE-LOADING T1CH & TIMEOUT\
020214  042  124  061  EM50M:  .ASCIZ  \"T1" FLAG CLEARED BY LOADING T1LL\
020256  042  124  061  EM50N:  .ASCIZ  \"T1" FLAG NOT CLEARED BY LOADING T1CH\
020324  042  120  102  EM50S:  .ASCIZ  \"PB7" W/IN VIA NOT SET ON TIMER 1 TIMEOUT\
020376  042  120  102  EM50U:  .ASCIZ  \"PB7" NOT SET AFTER TIMER 1 TIMEOUT\
020442  042  120  102  EM50V:  .ASCIZ  \"PB7" NOT DRIVEN LOW BY LOADING T1CH\
020507  042  120  102  EM50W:  .ASCIZ  \"PB7" UNEXPECTEDLY MODIFIED BY TIMER 1\
020556  042  124  061  EM50X:  .ASCIZ  \"T1" NOT RESET AFTER BEING CLEARED\
020621  042  120  102  EM50Y:  .ASCIZ  \"PB7" PREMATURELY SET DURING T1 COUNTDOWN\
020673  042  120  102  EM50Z:  .ASCIZ  \"PB7" NOT SET AFTER SECOND CYCLE\
020734  042  124  062  EM51B:  .ASCIZ  \"T2" FLAG NOT CLEARED BY LOADING T2CH\
021002  042  124  062  EM51C:  .ASCIZ  \"T2" FLAG NOT CLEARED BY READING T2CL\
021050  126  111  101  EM51E:  .ASCIZ  \VIA'S T2CH NOT DECREMENTING\
021104  042  124  062  EM51F:  .ASCIZ  \"T2" FLAG NOT SET ON TIMER 2 TIMEOUT\
```

ERROR MESSAGES -- "EM___"

```
3391 021151  042  124  062  EM51G:  .ASCIZ  \"T2" FLAG CLEARED BY READING T2CH\
3392 021213  042  124  062  EM51L:  .ASCIZ  \"T2" FLAG NOT SET AFTER RE-LOADING T2CH & TIMEOUT\
3393 021275  042  124  062  EM51M:  .ASCIZ  \"T2" FLAG CLEARED BY LOADING T2LL\
3394 021337  042  124  062  EM51N:  .ASCIZ  \"T2" FLAG NOT CLEARED BY LOADING T2CH\
3395 021405  042  124  062  EM51P:  .ASCIZ  \"T2" FLAG NOT SET AFTER APPROPRIATE DELAY\
3396 021457  042  123  122  EM52A:  .ASCIZ  \"SR" FLAG SET BEFORE ACCESSING SHIFT REGISTER\
3397 021535  116  117  040  EM52B:  .ASCIZ  \NO "SR" INT. USING MODE 2\
3398 021567  111  116  103  EM52C:  .ASCIZ  \INCOMPLETE SHIFTING OPERATION IN MODE 2 -- GOT INT.\
3399 021653  116  117  040  EM52D:  .ASCIZ  \NO "SR" INT. AFTER READING SR\
3400 021711  104  115  126  EM60N:  .ASCIZ  /DMV EXTENDED NPR WRITE ERROR/
3401
3402                                .EVEN
```

```
3404                                    .SBTTL  TEXT ADDRESS TABLES FOR ERROR HANDLERS -- "TXT__T"
3405                                    ;-------------------------------------------------------------------
3406                                    ;------------ TEXT ADDRESS TABLES USED BY ERROR HANDLERS ------------
3407                                    ;-------------------------------------------------------------------
3408
3409 021746  015515  015521  015535  TXTMLT: .WORD   TXTML0,TXTML1,TXTML2,TXTML3,TXTML4,TXTML5,TXTML6,TXTML7
3410
3411 021756  015745                          .WORD   TXTNP
3412 021770  015752  015762  015772  TXTNPT: .WORD   TXTNP0,TXTNP1,TXTNP2,TXTNP3,TXTNP4,TXTNP5,TXTNP6,TXTNP7,TXTNP8
3413
3414                                    .LIST   BEX
```

LOAD DEVICE PROTECTION TABLE

```
                                .SBTTL LOAD DEVICE PROTECTION TABLE

                                ;////////////////////////////////////////////////////////////////////////////////
                                ;/ THIS TABLE IDENTIFIES THE LOAD DEVICE TO THE SUPERVISOR, SO THAT IT CAN BE
                                ;/ PROTECTED FROM TESTING. IF DESIRED.
                                ;////////////////////////////////////////////////////////////////////////////////

022012                                  BGNPROT
022012                                                                  L$PROT::
022012  177777                          .WORD   -1      ;DON'T CHK CSR ADRS
022014  177777                          .WORD   -1      ;DON'T CHK MASSBUS UNIT NO.
022016  177777                          .WORD   -1      ;DON'T CHK DRIVE NO.
022020                                  ENDPROT
```

```
                        .SBTTL INITIALIZE SECTION

                        ;////////////////////////////////////////////////////////////////
                        ;/ THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                        ;/ AT THE BEGINNING OF THE TEST SEQUENCE ON THE NEXT UNIT.
                        ;////////////////////////////////////////////////////////////////

022020                          BGNINIT
022020                                                                  L$INIT::

022020                          SETVEC  #140,#170000,#340       ;ODT ROM ADDRESS                ;JB REV A-0
022020  012746  000340                                                  MOV     #340,-(SP)
022024  012746  170000                                                  MOV     #170000,-(SP)
022030  012746  000140                                                  MOV     #140,-(SP)
022034  012746  000003                                                  MOV     #3,-(SP)
022040  104437                                                          TRAP    C$SVEC
022042  062706  000010                                                  ADD     #10,SP

022046  010637  002270          MOV     SP,PSTACK       ;SAVE BASE-LEVEL STACK POINTER
                        ;SEE IF PROGRAM JUST STARTED, BR IF YES
022052                          READEF  #EF.START
022052  012700  000040                                                  MOV     #EF.START,R0
022056  104447                                                          TRAP    C$REFG
022060                          BCOMPLETE       STARST
022060  103417                                                          BCS     STARST

                        ;SEE IF PROGRAM JUST RESTARTED, BR IF YES
022062                          READEF  #EF.RESTART
022062  012700  000037                                                  MOV     #EF.RESTART,R0
022066  104447                                                          TRAP    C$REFG
022070                          BCOMPLETE       RESTRT
022070  103454                                                          BCS     RESTRT

                        ;SEE IF THIS IS A NEW PASS, BR IF YES
022072                          READEF  #EF.NEW
022072  012700  000035                                                  MOV     #EF.NEW,R0
022076  104447                                                          TRAP    C$REFG
022100                          BCOMPLETE       NEWST
022100  103454                                                          BCS     NEWST

                        ;SEE IF PROGRAM WAS JUST CONTINUED
022102                          READEF  #EF.CONTINUE
022102  012700  000036                                                  MOV     #EF.CONTINUE,R0
022106  104447                                                          TRAP    C$REFG
022110                          BCOMPLETE       10$
022110  103401                                                          BCS     10$
022112  000461                  BR      GETPRM

022114  000137  022414  10$:    JMP     CONTIN          ;(THIS IS TO FAR AWAY FOR A "BR" INSTRUCTION)

                        ;*** ENTER HERE IF "START" COMMAND ISSUED

022120  005037  002306  STARST: CLR     STARES          ;CLEAR FLAG TO SHOW JUST HAD STA OR RES

                        ; TEST FOR THE PRESENCE OR ABSENCE OF A CONSOLE TERMINAL.

022124  005037  002314          CLR     CONSOL          ;RESET THE CONSOLE TERMINAL FLAG
022130                          SETVEC  #4,#CONTST,#0   ;SETUP BUS TIMEOUT VECTER TO TEST FOR A CONSOLE
022130  012746  000000                                                  MOV     #0,-(SP)
```

INITIALIZE SECTION

```
        022134  012746  022514                                                  MOV     @CONTST,-(SP)
        022140  012746  000004                                                  MOV     #4,-(SP)
        022144  012746  000003                                                  MOV     #3,-(SP)
        022150  104437                                                          TRAP    C$SVEC
        022152  062706  000010                                                  ADD     #10,SP
3466    022156  005737  177564          TST     @#177564        ;TRY TO ACCESS THE CONSOLE TERMINAL'S "XCSR"
3467    022162                          CLRVEC  #4              ;WE SHOULD BE THROUGH WITH THIS BY NOW
        022162  012700  000004                                                  MOV     #4,R0
        022166  104436                                                          TRAP    C$CVEC
3468
3469                            ;  AT THIS POINT, IF A CONSOLE TERMINAL IS PRESENT, "CONSOL" WILL BE ZERO.
3470                            ;  IF NO CONSOLE TERMINAL EXISTS (OR AT LEAST NOT AT THE STANDARD ADDRESS),
3471                            ;       "CONSOL" WILL BE NON-ZERO (-1).
3472
3473    022170  005737  002314          TST     CONSOL          ;IF CONSOLE TERMINAL ISN'T THERE,
3474    022174  001412                  BEQ     5$
3475    022176                          PRINTF  #CFMTO,#NPROTS  ;TELL THE OPERATOR WHAT TESTING WON'T BE DONE
        022176  012746  000004                                                  MOV     #NPROTS,-(SP)
        022202  012746  022524                                                  MOV     #CFMTO,-(SP)
        022206  012746  000002                                                  MOV     #2,-(SP)
        022212  010600                                                          MOV     SP,R0
        022214  104417                                                          TRAP    C$PNTF
        022216  062706  000006                                                  ADD     #6,SP
3476    022222                  5$:
3477                            ;*** ENTER HERE IF "RESTART" COMMAND ISSUED
3478
3479    022222  005037  002306  RESTRT: CLR     STARES          ;CLEAR FLAG TO SHOW JUST HAD STA OR RES
3480    022226  005037  002310          CLR     DEVMAP          ;CLEAR DEVICE MAP
3481
3482    022232                  NEWST:          ;ENTER HERE BEFORE EACH TEST
3483
3484    022232  012737  177777  002266          MOV     #-1,LOGDEV      ;RESET LOGICAL DEVICE TO -1
3485    022240  005237  002306                  INC     STARES          ;INC # OF PASSES SINCE STA OR RES
3486    022244  005237  002304                  INC     FRSPAS          ;INCREMENT NO. OF PASSES AFTER LOAD
3487    022250  012737  000001  002312          MOV     #BIT0,DEVPTR    ;INIT DEVICE MAP BIT POINTER
3488                    ; GET UNIBUS ADDRESS, VECTOR, PRIORITY LEVEL, SWITCH PACKS, TEST
3489                    ;   CONNECTOR INFORMATION FOR THIS LOGICAL DEVICE
3490    022256          GETPRM:
3491    022256  005237  002266          INC     LOGDEV          ;INCREMENT LOGICAL DEVICE NUMBER
3492    022262                          GPHARD  LOGDEV,R1       ;GET P-TABLE POINTER INTO R1
        022262  013700  002266                                                  MOV     LOGDEV,R0
        022266  104442                                                          TRAP    C$GPHRD
        022270  010001                                                          MOV     R0,R1
3493    022272                          BCOMPLETE  10$          ;BR IF DEVICE AVAILABLE
        022272  103403                                                          BCS     10$
3494    022274  006337  002312          ASL     DEVPTR          ;IF UN-AVAILABLE, SHIFT DEVICE MAP BIT POINTER
3495    022300  000766                  BR      GETPRM          ;       AND SKIP THIS DEVICE
3496
3497    022302  053737  002312  002310  10$:    BIS     DEVPTR,DEVMAP   ;ELSE, SET BIT FOR THIS DEVICE IN DEVICE MAP
3498    022310  006337  002312          ASL     DEVPTR          ;SHIFT DEVICE MAP BIT POINTER
3499
3500                            ;  "R1" WAS RETURNED WITH A POINTER TO THE CURRENT "P-TABLE"
3501
3502    022314  012100                  MOV     (R1)+,R0        ;GET THE DEVICE CSR ADDRESS
3503    022316  012703  000020          MOV     #16.,R3         ;WE HAVE TO SETUP THIS MANY ADDRESS POINTERS
3504    022322  012702  002320          MOV     #MPCSR,R2       ;THIS IS THE ADDRESS OF THE FIRST POINTER
3505    022326  010022          12$:    MOV     R0,(R2)+        ;SETUP ONE CSR POINTER
```

INITIALIZE SECTION

```
3506 022330  005200                        INC     R0              ;POINT TO THE NEXT CSR ADDRESS
3507 022332  077303                        SOB     R3,12$          ;LOOP AS LONG AS THERE ARE MORE TABLE ENTRIES
3508                                                               ;ELSE,  FALL THROUGH TO CONTINUE GETTING MORE
3509                                                               ;       P-TABLE DATA
3510
3511 022334  012100                        MOV     (R1)+,R0        ;GET INTERRUPT VECTOR
3512 022336  010037  002360                MOV     R0,MPIVEC       ;SETUP "A" VECTOR POINTER
3513 022342  022020                        CMP     (R0)+,(R0)+     ;ADD 4 TO VECTOR TO GET ADDRESS OF "B" VECTOR
3514 022344  010037  002362                MOV     R0,MPOVEC       ;SETUP "B" VECTOR POINTER
3515
3516 022350  012100                        MOV     (R1)+,R0        ;GET DMV11 DEVICE PRIORITY
3517 022352  006200                        ASR     R0              ;       RE-POSITION IT
3518 022354  006200                        ASR     R0
3519 022356  006200                        ASR     R0
3520 022360  006200                        ASR     R0
3521 022362  010037  002364                MOV     R0,MPRIOR       ;SETUP OUR VARIABLE FOR INT. VECTOR INIT'S
3522
3523 022366  022121                        CMP     (R1)+,(R1)+     ;SKIP OVER SWITCH #'S 1 & 2
3524 022370  012137  002366                MOV     (R1)+,BRDTYP    ;GET DMV-11 BOARD TYPE
3525 022374  022111                        CMP     (R1)+,(R1)      ;SKIP OVER CONNECTOR FLAG
3526
3527 022376  012100                        MOV     (R1)+,R0        ;GET CONTROL FLAGS
3528 022400  012703  000006                MOV     #6,R3           ;POSITION THEM PROPERLY IN THE WORD
3529 022404  006200              15$:      ASR     R0
3530 022406  077302                        SOB     R3,15$
3531 022410  010037  002370                MOV     R0,PT.CTL       ;PUT IT WHERE TESTS EXPECT TO FIND IT
3532
3533                                ;;  TEST THE VARIOUS CONTROL FLAGS & REPORT NON-STANDARD ACTION RESULTING FROM  ;JB REV A-0
3534                                ;;  THEIR SETTINGS                                                             ;JB REV A-0
3535                                ;                                                                              ;JB REV A-0
3536                                ;       CMP     STARES,#1       ;FIRST PASS SINCE STA OR RES ??                ;JB REV A-0
3537                                ;       BNE     40$             ; IF NO: SKIP POSSIBLE PRINTOUT                ;JB REV A-0
3538                                ;                                                                              ;JB REV A-0
3539                                ;       BIT     #PU24,PT.CTL    ;IF THE PROCESSOR ISN'T STRAPPED TO COME UP    ;JB REV A-0
3540                                ;       BNE     40$             ;THROUGH INTERRUPT VECTOR 24 & 26.             ;JB REV A-0
3541                                ;       PRINTF  #CFMT2,#DCOKTS,#HLTEST ;TELL THE OPERATOR THAT NO DCOK TESTING ;JB REV A-0
3542                                ;       PRINTF  #CFMT3          ;WILL BE DONE                                  ;JB REV A-0
3543 022414                        40$:
3544
3545 022414                        CONTIN:         ;ENTER HERE WHEN A "CONTINUE" COMMAND IS ISSUED
3546
3547 022414                                SETVEC  @#MPIVEC,#MPIHAN,@#MPRIOR  ;SETUP "A" INT. VECTOR
     022414  013746  002364                                                            MOV     @#MPRIOR,-(SP)
     022420  012746  006066                                                            MOV     #MPIHAN,-(SP)
     022424  013746  002360                                                            MOV     @#MPIVEC,-(SP)
     022430  012746  000003                                                            MOV     #3,-(SP)
     022434  104437                                                                    TRAP    C$SVEC
     022436  062706  000010                                                            ADD     #10,SP
3548 022442  005037  006136                CLR     IHILNK          ;WE DON'T WANT THE HANDLER TO LINK ELSEWHERE
3549 022446                                SETVEC  @#MPOVEC,#MPOHAN,@#MPRIOR  ;SETUP "B" INT. VECTOR
     022446  013746  002364                                                            MOV     @#MPRIOR,-(SP)
     022452  012746  006140                                                            MOV     #MPOHAN,-(SP)
     022456  013746  002362                                                            MOV     @#MPOVEC,-(SP)
     022462  012746  000003                                                            MOV     #3,-(SP)
     022466  104437                                                                    TRAP    C$SVEC
     022470  062706  000010                                                            ADD     #10,SP
3550 022474  005037  006210                CLR     IHOLNK          ;WE DON'T WANT THE HANDLER TO LINK ELSEWHERE
```

INITIALIZE SECTION

```
3551 022500  005037  002274                CLR     INTWCH          ;RESET "INTERRUPT WATCH" FLAGS (BOTH "A" & "B")
3552
3553 022504  012737  000001  002302        MOV     #1,FRSTIM       ;MARK FLAG FOR NEXT TIME THROUGH
3554 022512                                ENDINIT                 ;END OF "INIT" CODE
     022512                                                                L10022:
     022512  104411                                                            TRAP    C$INIT
3555
3556                                ; ***** SUBROUTINES USED BY "INIT" CODE *****
3557
3558                                ;       INTERRUPT HANDLER FOR CONSOLE TERMINAL PRESENCE TESTING
3559
3560 022514  012737  177777  002314  CONTST: MOV     #-1,CONSOL      ;INDICATE THAT NO CONSOLE TERMINAL EXISTS!
3561 022522  000002                        RTI                     ;RETURN
3562
3563                                ;       FORMATS FOR FORCED MESSAGES
3564                                .NLIST  BEX
3565 022524      045      116      045  CFMT0:  .ASCIZ  /%N%A TEST %D2%A SUBTEST 3 CAN'T RUN   NO CSR @ 177564/
3566 022613      045      116      045  CFMT2:  .ASCIZ  /%N%A TESTS %D2%A AND %D2%A CAN'T RUN   CPU NOT/
3567 022673      045      116      045  CFMT3:  .ASCIZ  /%N%S4%ATRAPPED TO POWER-UP THROUGH VECTOR 24/
3568                                .LIST   BEX
3569                                .EVEN
```

```
                        .SBTTL AUTO DROP UNIT SECTION

                        ;//////////////////////////////////////////////////////////////////////////////
                        ;/ THE AUTO DROP CODING DETERMINES WHETHER OR NOT THE DEVICE WHOSE P-TABLE
                        ;/ WAS JUST OBTAINED IS READY FOR TESTING, AND IT IS DROPPED IF NOT READY.
                        ;//////////////////////////////////////////////////////////////////////////////

                        ;+*****************************************************************************
                        ;
                        ;       THIS ALGORITHM IS THE SAME A CNDMA TEST # 1 EXCEPT THAT TEST
                        ;       WILL JUST REPORT THE FAILURE AND GO ON -- THIS ROUTINE WILL CAUSE THE
                        ;       DEVICE TO BE DROPPED IF A BUS-TIMEOUT OCCURS WHEN ANY OF THE CSR'S
                        ;       ARE ACCESSED WITH EITHER A "TST" OR "TSTB" INSTRUCTION.
                        ;
                        ;-*****************************************************************************

                                BGNAUTO
                                                                        L$AUTO::
                                SETVEC  #4,#AD.HIT,#0   ;SETUP INVALID-ADDRESS TRAP VECTOR
                                                                                MOV     #0,-(SP)
                                                                                MOV     #AD.HIT,-(SP)
                                                                                MOV     #4,-(SP)
                                                                                MOV     #3,-(SP)
                                                                                TRAP    C$SVEC
                                                                                ADD     #10,SP
                                CLR     TMP0            ;INITIALIZE TRAP FLAG REGISTER
                                MOV     #1,R2           ;FLAG BIT
                                MOV     BSELO,R3        ;INIT ADDRESS POINTER

                        1$:     TSTB    (R3)+           ;ACCESS THE CSR'S BY BYTES.
                                ASL     R2
                                BCC     1$

                                MOV     BSELO,R3        ;RE-INIT ADDRESS POINTER
                                MOV     #1,R2           ;RE-INIT FLAG BIT
                        2$:     TST     (R3)+           ;ACCESS THE CSR'S BY WORDS.
                                ASL     R2
                                ASL     R2
                                BCC     2$

                                CLRVEC  #4              ;RESTORE THE VECTOR TO DS
                                                                                MOV     #4,R0
                                                                                TRAP    C$CVEC
                                TST     TMP0            ;DID WE GET HIT WITH AN INVALID ADDRESS TRAP?
                                BEQ     AD.OK           ;NO,  EXIT TEST
                                DODU    LOGDEV          ;YES, DROP THIS LOGICAL DEV.
                                                                                MOV     LOGDEV,R0
                                                                                TRAP    C$DODU

                        AD.OK:  NOP                     ;(FOR PATCHING IN A HALT IF NECESSARY)

                                ENDAUTO
                                                                        L10023:
                                                                                TRAP    C$AUTO

                        AD.HIT: BIS     R2,TMP0         ;FLAG THE HIT IF WE GET IT!
```

AUTO DROP UNIT SECTION

```
3615 023074  000002                    RTI                    ;RETURN
3616
```

CLEANUP CODING SECTION

```
                                .SBTTL CLEANUP CODING SECTION

                                ;////////////////////////////////////////////////////////////////
                                ;/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ;/ AT THE END OF THE TEST SEQUENCE ON A PARTICULAR UNIT.
                                ;////////////////////////////////////////////////////////////////

                                        BGNCLN
023076                                                                  L$CLEAN::
                                        CLRVEC  @#MPIVEC        ;RETURN VECTORS TO SUPERVISOR
023076  013700  002360                                                          MOV     @#MPIVEC,R0
023102  104436                                                                 TRAP    C$CVEC
                                        CLRVEC  @#MPOVEC
023104  013700  002362                                                          MOV     @#MPOVEC,R0
023110  104436                                                                 TRAP    C$CVEC
                                        ENDCLN
023112                                                                  L10024:
023112  104412                                                                 TRAP    C$CLEAN
```

DROP UNIT SECTION

```
                                .SBTTL DROP UNIT SECTION

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

023114                                  BGNDU
023114                                                                  L$DU::
                                ;ISSUE UNIBUS RESET TO CLEAN UP
023114                                  BRESET
023114  104433                                                                  TRAP    C$RESET
023116                                  ENDDU
023116                                                                  L10025:
023116  104453                                                                  TRAP    C$DU
```

ADD UNIT SECTION

```
                                .SBTTL ADD UNIT SECTION
                                ;////////////////////////////////////////////////////////////////
                                ;/ THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ;/  TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
                                ;/  "EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
                                ;////////////////////////////////////////////////////////////////

023120                                  BGNAU
023120                                                                  L$AU::
023120                                  ENDAU
023120                                                                  L10026:
023120  104452                                                                  TRAP    C$AU
```

```
                        .SBTTL TEST 1 -- VIA TIMER 2 ONE SHOT MODE

                        ;******************************************************************************
                        ;*
                        ;*      TEST 1 -- VIA TIMER 2 ONE SHOT MODE
                        ;*
                        ;*    THIS  TEST  VERIFIES  THAT  THE  TIMER 2 COUNTER IS OPERATIONAL IN
                        ;*    INTERVAL-TIMER (ONE-SHOT) MODE.
                        ;*
                        ;*      THE FOLLOWING IS PERFORMED :
                        ;*
                        ;*    A  MASTER CLEAR IS DONE & THE TIMER IS PLACED IN INTERVAL-TIMER MODE
                        ;*    BY  SETTING  ACR5 = 0 AND THE PROGRAM CHECKS FOR "T2" (BIT 5 IN IFR)
                        ;*    TO BE INITIALLY CLEARED.
                        ;*
                        ;*    T2L-L  (ADR  08)  & T2C-H (ADR 09) ARE BOTH LOADED WITH 252 (OCTAL).
                        ;*    (THIS  IS  EQUIVALENT  TO  AAAA  (HEX) OR 43,690 (DECIMAL).) LOADING
                        ;*    T2C-H STARTS THE COUNTER.
                        ;*
                        ;*    T2L-L  IS  LOADED  WITH 001 AND T2C-H IS LOADED WITH 000 IN ORDER TO
                        ;*    SET  "T2"  WITH  A QUICK UNDERFLOW. THE "T2" FLAG BIT IN IFR IS READ
                        ;*    AND CHECKED TO BE SET.
                        ;*
                        ;*    T2C-H IS CHECKED TO = 0. CHECKING T2C-H SHOULD NOT HAVE CLEARED "T2"
                        ;*    -- THIS IS VERIFIED.
                        ;*
                        ;*    T2C-L  IS CHECKED TO = 0. CHECKING T2C-L SHOULD HAVE CLEARED "T2" --
                        ;*    THIS TOO IS VERIFIED.
                        ;*
                        ;*    T2C-H  IS  LOADED  WITH  0 AGAIN TO INITIATE A NEW COUNT DOWN (WHICH
                        ;*    SHOULD  UNDERFLOW  ALMOST  IMMEDIATELY)  AND  THE "T2" BIT IN IFR IS
                        ;*    CHECKED TO BE SET AGAIN.
                        ;*
                        ;*    T2L-L IS LOADED WITH 125 (OCTAL) AND "T2" BIT IS CHECKED TO BE STILL
                        ;*    SET.
                        ;*
                        ;*    T2C-H IS LOADED WITH 125, AND THE "T2" BIT IS READ AND CHECKED TO BE
                        ;*    CLEARED BY THE LOADING OF T2C-H.
                        ;*
                        ;*
                        ;******************************************************************************
                        ;
                        ;       BGNTST
                        ;
023122                  ;       BGNSUB                                          T1::
023122                  ;                                                       T1.1:
023122  104402          ;                                                               TRAP    C$BSUB
                        ;-------------------------------------------------------------------------------
                        ;
                        ;       JSR     PC,MSTCLR       ;INIT DMV & ENTER M-LOOP
023124  004737  003514  ;       BCC     1$              ;IF NO ERROR, PROCEED WITH TESTING
023130  103003          ;       ERROR                   ;ELSE, REPORT ERROR
023132                  ;                                                               TRAP    C$ERROR
023132  104460          ;       ESCAPE  TST             ;   & EXIT TEST
023134                  ;                                                               TRAP    C$ESCAPE
023134  104410                                                                          .WORD   L100??
023136  000700
```

SEQ 0099

TEST 1 -- VIA TIMER 2 ONE SHOT MODE

```
3699 023140  004537  004630          1$:     JSR     R5,INITT2       ;INITIALIZE TIMER # 2
3700 023144  002000                          2000                    ;  2000 ==> LATCHES  (PREVENTS IMMED. TIMEOUT)
3701 023146  000000                          0                       ;  MODE 0 & "T2" INT. ENABLE FLAG CLEARED
3702 023150  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
3703 023152                                  ERROR                   ;ELSE, REPORT IT
     023152  104460                                                                  TRAP    C$ERROR
3704 023154                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     023154  104410                                                                  TRAP    C$ESCAPE
     023156  000660                                                                  .WORD   L10027-.
3705 023160  004737  024072                  JSR     PC,GETT2        ;IS "T2" SET?
3706 023164  102002                          BVC     .+6             ;IF NO ERROR, PROCEED
3707 023166                                  ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023166  104410                                                                  TRAP    C$ESCAPE
     023170  000644                                                                  .WORD   L10030-.
3708 023172  103033                          BCC     6$              ;NO, GOOD.
3709 023174                                  GEDF    EM51B,ERR51     ;YES, REPORT IT'S NOT BEING CLEARED & INIT.
                                                                     ;       "DEVICE FATAL" ERROR # 9
     023174  104455                                                                  TRAP    C$ERDF
     023176  000011                                                                  .WORD   9
     023200  020734                                                                  .WORD   EM51B
     023202  011176                                                                  .WORD   ERR51
3710
3711                                 ;---------- ---------- ----------
3712
3713 023204  112737  000002  002435          MOVB    #2,TMP9+1
3714 023212  004537  004042                  JSR     R5,WRITE        ;INIT TIMER # 2 BY WRITING INTO
3715 023216  120011                          T2CH                    ;T2C-H (ADDR 09)
3716 023220  002435                          TMP9+1
3717 023222  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
3718 023224                                  ERROR                   ;ELSE, REPORT IT
     023224  104460                                                                  TRAP    C$ERROR
3719 023226                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     023226  104410                                                                  TRAP    C$ESCAPE
     023230  000606                                                                  .WORD   L10027-.
3720 023232  004737  024072                  JSR     PC,GETT2        ;IS "T2" SET?
3721 023236  102002                          BVC     .+6             ;IF NO ERROR, PROCEED
3722 023240                                  ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023240  104410                                                                  TRAP    C$ESCAPE
     023242  000572                                                                  .WORD   L10030-.
3723 023244  103006                          BCC     6$              ;NO, GOOD.
3724 023246                                  GEDF    EM51B,ERR51     ;YES, REPORT IT'S NOT BEING CLEARED & INIT.
                                                                     ;       "DEVICE FATAL" ERROR # 10
     023246  104455                                                                  TRAP    C$ERDF
     023250  000012                                                                  .WORD   10
     023252  020734                                                                  .WORD   EM51B
     023254  011176                                                                  .WORD   ERR51
3725 023256                                  ESCAPE  SUB             ;AND EXIT SUBTEST
     023256  104410                                                                  TRAP    C$ESCAPE
     023260  000554                                                                  .WORD   L10030-.
3726
3727                                 ;---------- ---------- ---------- ---------- ----------
3728
3729 023262  004537  024040          6$:     JSR     R5,LODT2C       ;LOAD TIMER # 2
3730 023266     252                  7$:     .BYTE   252
3731 023267     252                  8$:     .BYTE   252
3732
3733                                 ;---------- ---------- ---------- ---------- ----------
```

TEST 1 -- VIA TIMER 2 ONE SHOT MODE

```
3734
3735 023270  004537  003616          JSR     R5,READ         ;READ THE LOW COUNTER
3736 023274  120010                  T2CL
3737 023276  002432                  TMP8
3738 023300  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
3739 023302                          ERROR                   ;ELSE, REPORT IT
     023302  104460                                                  TRAP    C$ERROR
3740 023304                          ESCAPE  TST             ;       AND EXIT THIS TEST
     023304  104410                                                  TRAP    C$ESCAPE
     023306  000530                                                  .WORD   L10027-.
3741 023310  123737  002432  023266  CMPB    TMP8,7$         ;MAKE SURE THE COUNTER IS DECREMENTING
3742 023316  001004                  BNE     12$             ;IT IS, NOW SEE IF THE HIGH COUNTER IS TOO
3743 023320                          GEDF    EM50D,ERR51     ;IT WASN'T -- REPORT THE ERROR
                                                             ;       "DEVICE FATAL" ERROR # 11
     023320  104455                                                  TRAP    C$ERDF
     023322  000013                                                  .WORD   11
     023324  017453                                                  .WORD   EM50D
     023326  011176                                                  .WORD   ERR51
3744 023330  012703  000100  12$:   MOV     #100,R3         ;INIT. TIMEOUT VALUE
3745 023334  004537  003616  13$:   JSR     R5,READ         ;READ THE HIGH COUNTER
3746 023340  120011                  T2CH
3747 023342  002434                  TMP9
3748 023344  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
3749 023346                          ERROR                   ;ELSE, REPORT IT
     023346  104460                                                  TRAP    C$ERROR
3750 023350                          ESCAPE  TST             ;       AND EXIT THIS TEST
     023350  104410                                                  TRAP    C$ESCAPE
     023352  000464                                                  .WORD   L10027-.
3751 023354  123737  002434  023267  CMPB    TMP9,8$         ;DID IT CHANGE FROM THE LOADED VALUE?
3752 023362  001007                  BNE     14$             ;YES, PROCEED WITH TESTING
3753 023364  077315                  SOB     R3,13$          ;NO,  IF NO TIMEOUT, TRY AGAIN
3754 023366                          GEDF    EM51E,ERR51     ;ELSE, REPORT THAT HIGH COUNTER ISN'T RUNNING
                                                             ;       "DEVICE FATAL" ERROR # 12
     023366  104455                                                  TRAP    C$ERDF
     023370  000014                                                  .WORD   12
     023372  021050                                                  .WORD   EM51E
     023374  011176                                                  .WORD   ERR51
3755 023376                          ESCAPE  SUB             ;       WE CAN'T PROCEED WITH TESTING EITHER
     023376  104410                                                  TRAP    C$ESCAPE
     023400  000434                                                  .WORD   L10030-.
3756                         ;-------------------------------------------------------------------------
3757
3758 023402  005003          14$:   CLR     R3              ;INITIALIZE TIMEOUT COUNTER
3759 023404  004737  024072  15$:   JSR     PC,GETT2        ;WAIT FOR TIMER TO COUNT DOWN
3760 023410  102002                  BVC     .+6             ;IF NO ERROR, PROCEED
3761 023412                          ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023412  104410                                                  TRAP    C$ESCAPE
     023414  000420                                                  .WORD   L10030-.
3762 023416  103406                  BCS     16$             ;DONE.
3763 023420  077307                  SOB     R3,15$          ;NOT YET, TIMEOUT?
3764 023422                          GEDF    EM51P,ERR51     ;YES, REPORT NO "T2" INT. FLAG
                                                             ;       "DEVICE FATAL" ERROR # 13
     023422  104455                                                  TRAP    C$ERDF
     023424  000015                                                  .WORD   13
     023426  021405                                                  .WORD   EM51P
     023430  011176                                                  .WORD   ERR51
3765 023432  000445                  BR      17$             ;     & BYPASS "T2"-RESET-ON-T2CH-READ CHECK
```

```
3766
3767                    ;-------------------------------------------------------------------------
3768
3769 023434  004537  003616      16$:    JSR     R5,READ         ;READ T2C-H (ADDR 09)
3770 023440  120011                      T2CH
3771 023442  002434                      TMP9
3772 023444  103003                      BCC     .+10            ;IF NO ERROR, PROCEED
3773 023446                              ERROR                   ;ELSE, REPORT IT
     023446  104460                                                      TRAP    C$ERROR
3774 023450                              ESCAPE  TST             ;       AND EXIT THIS TEST
     023450  104410                                                      TRAP    C$ESCAPE
     023452  000364                                                      .WORD   L10027-.
3775 023454  004737  024072              JSR     PC,GETT2        ;IS "T2" STILL SET?
3776 023460  102002                      BVC     .+6             ;IF NO ERROR, PROCEED
3777 023462                              ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023462  104410                                                      TRAP    C$ESCAPE
     023464  000350                                                      .WORD   L10030-.
3778 023466  103405                      BCS     40$             ;YES, ALL'S OK
3779 023470                              GEDF    EM51G,ERR51     ;NO!  BAD VIA CHIP!
                                                                 ;       "DEVICE FATAL" ERROR # 14
     023470  104455                                                      TRAP    C$ERDF
     023472  000016                                                      .WORD   14
     023474  021151                                                      .WORD   EM51G
     023476  011176                                                      .WORD   ERR51
3780 023500  000422                      BR      17$             ;   & BYPASS "T2"-RESET-ON-T2LL-WRITE CHECK
3781
3782                    ;-------------------------------------------------------------------------
3783
3784 023502  004537  004042      40$:    JSR     R5,WRITE        ;RE-LOAD T2L-L (ADDR 08)
3785 023506  120010                      T2LL
3786 023510  002433                      TMP8+1
3787 023512  103003                      BCC     .+10            ;IF NO ERROR, PROCEED
3788 023514                              ERROR                   ;ELSE, REPORT IT
     023514  104460                                                      TRAP    C$ERROR
3789 023516                              ESCAPE  TST             ;       AND EXIT THIS TEST
     023516  104410                                                      TRAP    C$ESCAPE
     023520  000316                                                      .WORD   L10027-.
3790 023522  004737  024072              JSR     PC,GETT2        ;IS "T2" STILL SET?
3791 023526  102002                      BVC     .+6             ;IF NO ERROR, PROCEED
3792 023530                              ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023530  104410                                                      TRAP    C$ESCAPE
     023532  000302                                                      .WORD   L10030-.
3793 023534  103404                      BCS     17$             ;YES, ALL'S STILL OK
3794 023536                              GEDF    EM51M,ERR51     ;NO!  SOMETHING WENT WRONG!  REPORT IT
                                                                 ;       "DEVICE FATAL" ERROR # 15
     023536  104455                                                      TRAP    C$ERDF
     023540  000017                                                      .WORD   15
     023542  021275                                                      .WORD   EM51M
     023544  011176                                                      .WORD   ERR51
3795
3796                    ;-------------------------------------------------------------------------
3797
3798 023546  004537  024040      17$:    JSR     R5,LODT2C       ;RE-LOAD TIMER # 2 WITH A VALUE WHICH CAUSE AN
3799 023552     001              18$:    .BYTE   1               ;ALMOST IMMEDIATE TIMEOUT
3800 023553     000              19$:    .BYTE   0               ; (ADDRESS OF HIGH BYTE FOR T2C-H (ADDR 09))
3801                    ;-------------------------------------------------------------------------
3802
```

TEST 1 -- VIA TIMER 2 ONE SHOT MODE

```
3803 023554  004737  024072          JSR     PC,GETT2        ;WAS "T2" SET BY THE ABOVE OPERATION?
3804 023560  102002                  BVC     .+6             ;IF NO ERROR, PROCEED
3805 023562                          ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023562  104410                                                          TRAP    C$ESCAPE
     023564  000250                                                          .WORD   L10030-.
3806 023566  103406                  BCS     20$             ;YES, OK -- CONTINUE ERROR CHECKING
3807 023570                          GEDF    EM51F,ERR51     ;NO,  BAD NEWS!  REPORT THE FAILURE
                                                             ;       "DEVICE FATAL" ERROR # 16
     023570  104455                                                          TRAP    C$ERDF
     023572  000020                                                          .WORD   16
     023574  021104                                                          .WORD   EM51F
     023576  011176                                                          .WORD   ERR51
3808 023600                          ESCAPE  SUB             ;  AND GET OUT OF SUBTEST
     023600  104410                                                          TRAP    C$ESCAPE
     023602  000232                                                          .WORD   L10030-.
3809 023604  004537  003616  20$:    JSR     R5,READ         ;READ T2C-H (ADDR 09) TO SEE IF THIS CLEARS "T2"
3810 023610  120011                  T2CH                    ;(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE
3811 023612  002434                  TMP9                    ; ALMOST ANYTHING)
3812 023614  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
3813 023616                          ERROR                   ;ELSE, REPORT IT
     023616  104460                                                          TRAP    C$ERROR
3814 023620                          ESCAPE  TST             ;       AND EXIT THIS TEST
     023620  104410                                                          TRAP    C$ESCAPE
     023622  000214                                                          .WORD   L10027-.
3815 023624  004737  024072          JSR     PC,GETT2        ;PUT THE CURRENT "T2" VALUE INTO THE CARRY BIT
3816 023630  102002                  BVC     .+6             ;IF NO ERROR, PROCEED
3817 023632                          ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023632  104410                                                          TRAP    C$ESCAPE
     023634  000200                                                          .WORD   L10030-.
3818 023636  103405                  BCS     21$             ;IF SET, READING T2CH DIDN'T CLEAR IT -- OK!
3819 023640                          GEDF    EM50G,ERR51     ;IF CLEARED!  BAD VIA CHIP!
                                                             ;       "DEVICE FATAL" ERROR # 17
     023640  104455                                                          TRAP    C$ERDF
     023642  000021                                                          .WORD   17
     023644  017610                                                          .WORD   EM50G
     023646  011176                                                          .WORD   ERR51
3820 023650  000400                  BR      28$             ;BYPASS THE REST OF THIS SECTION OF TESTING
3821
3822 023652                  21$:
3823                         ;-------------------------------------------------------------------------- ---
3824
3825 023652  004537  003616  28$:    JSR     R5,READ         ;READ T2C-L (ADDR 08)
3826 023656  120010                  T2CL                    ;(THIS VALUE ISN'T CHECKED BECAUSE IT CAN BE
3827 023660  002432                  TMP8                    ; ALMOST ANYTHING)
3828 023662  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
3829 023664                          ERROR                   ;ELSE, REPORT IT
     023664  104460                                                          TRAP    C$ERROR
3830 023666                          ESCAPE  TST             ;       AND EXIT THIS TEST
     023666  104410                                                          TRAP    C$ESCAPE
     023670  000146                                                          .WORD   L10027-.
3831 023672  004737  024072          JSR     PC,GETT2        ;IS "T2" CLEARED NOW
3832 023676  102002                  BVC     .+6             ;IF NO ERROR, PROCEED
3833 023700                          ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023700  104410                                                          TRAP    C$ESCAPE
     023702  000132                                                          .WORD   L10030-.
3834 023704  103004                  BCC     29$             ;YES, ALL'S OK
3835 023706                          GEDF    EM51C,ERR51     ;NO!  BAD VIA CHIP!
```

TEST 1 -- VIA TIMER 2 ONE SHOT MODE

```
                                                        ;       "DEVICE FATAL" ERROR # 18
     023706  104455                                                                  TRAP    C$ERDF
     023710  000022                                                                  .WORD   18
     023712  021002                                                                  .WORD   EM51C
     023714  011176                                                                  .WORD   ERR51
3836
3837                            ;-------------------------------------------------------------------------------
3838
3839 023716  004537  004042     29$:    JSR     R5,WRITE        ;RE-WRITE INTO T2C-H (ADDR 09) TO SET T2 AGAIN
3840 023722  120011             T2CH
3841 023724  002435             TMP9+1
3842 023726  103003             BCC     .+10            ;IF NO ERROR, PROCEED
3843 023730                     ERROR                   ;ELSE, REPORT IT
     023730  104460                                                                  TRAP    C$ERROR
3844 023732                     ESCAPE  TST             ;       AND EXIT THIS TEST
     023732  104410                                                                  TRAP    C$ESCAPE
     023734  000102                                                                  .WORD   L10027-.
3845 023736  004737  024072     JSR     PC,GETT2        ;IS "T2" SET AGAIN
3846 023742  102002             BVC     .+6             ;IF NO ERROR, PROCEED
3847 023744                     ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     023744  104410                                                                  TRAP    C$ESCAPE
     023746  000066                                                                  .WORD   L10030-.
3848 023750  103406             BCS     32$             ;YES, ALL'S WELL (AGAIN?)
3849 023752                     GEDF    EM51L,ERR51     ;NO!  SOMETHING WENT WRONG!  REPORT IT
                                                        ;       "DEVICE FATAL" ERROR # 19
     023752  104455                                                                  TRAP    C$ERDF
     023754  000023                                                                  .WORD   19
     023756  021213                                                                  .WORD   EM51L
     023760  011176                                                                  .WORD   ERR51
3850 023762                             ESCAPE  SUB             ;       AND EXIT FROM THIS SUBTEST
     023762  104410                                                                  TRAP    C$ESCAPE
     023764  000050                                                                  .WORD   L10030-.
3851
3852                            ;-------------------------------------------------------------------------------
3853
3854 023766  004537  024040     32$:    JSR     R5,LODT2C       ;AGAIN RE-LOAD TIMER # 2.  THIS TIME WITH
3855 023772     125     125             .BYTE   125,125         ;  LARGER BUT DIFFERENT VALUES
3856
3857                            ;-------------------------------------------------------------------------------
3858
3859 023774  004737  024072             JSR     PC,GETT2        ;"T2" SHOULD NOW BE CLEARED
3860 024000  102002             BVC     .+6             ;IF NO ERROR, PROCEED
3861 024002                     ESCAPE  SUB             ;ELSE, IT'S ALREADY BEEN REPORTED -- EXIT
     024002  104410                                                                  TRAP    C$ESCAPE
     024004  000030                                                                  .WORD   L10030-.
3862 024006  103004             BCC     34$             ;IT WAS, ALL'S WELL THAT END'S WELL (I THINK!?)
3863 024010                     GEDF    EM51N.ERR51     ;IT WASN'T!  SOMETHING WENT WRONG!  REPORT IT
                                                        ;       "DEVICE FATAL" ERROR # 20
     024010  104455                                                                  TRAP    C$ERDF
     024012  000024                                                                  .WORD   20
     024014  021337                                                                  .WORD   EM51N
     024016  011176                                                                  .WORD   ERR51
3864
3865 024020  004537  004630     34$:    JSR     R5,INITT2       ;RE-INITIALIZE TIMER # 1 W/ NORMAL RESET VALUES
3866 024024  000000             0
3867 024026  000000             0
3868 024030  103001             BCC     .+4             ;IF NO ERROR, EXIT
```

TEST 1 -- VIA TIMER 2 ONE SHOT MODE

```
3869 024032                          ERROR                    ;ELSE, REPORT IT
     024032  104460                                                        TRAP    C$ERROR
3870
3871 024034                          ENDSUB
     024034                                                   L10030:
     024034  104403                                                        TRAP    C$ESUB
3872
3873 024036                          ENDTST
     024036                                                   L10027:
     024036  104401                                                        TRAP    C$ETST
3874
3875                          ;-------------------------------------------------------------------------
3876                            LODT2C -- LOAD TIMER TWO AT ADDRESSES 08 & 09
3877                          ;
3878                          ; CALLING SEQUENCE:
3879                          ;
3880                          ;       JSR     R5,LODT2C
3881                          ;       .BYTE   <VALUE FOR T2L-L (ADDRESS 08)>
3882                          ;       .BYTE   <VALUE FOR T2C-H (ADDRESS 09)>
3883                          ;       <NEXT SEQUENCIAL INSTRUCTION
3884                          ;
3885                          ;-------------------------------------------------------------------------
3886
3887 024040  112537  002433  LODT2C: MOVB    (R5)+,TMP8+1     ;SETUP TO LOAD T2LL
3888 024044  112537  002435          MOVB    (R5)+,TMP9+1     ;  AND T2CH
3889 024050  004537  004042          JSR     R5,WRITE         ;LOAD T2L-L (ADDR 08) WITH PASSED PARAMETER
3890 024054  120010                  T2LL
3891 024056  002433                  TMP8+1
3892 024060  004537  004042          JSR     R5,WRITE         ;LOAD T2C-H (ADDR 09) WITH PASSED PARAMETER
3893 024064  120011                  T2CH                     ;  (THIS WILL ALSO RESET "T2" & THE COUNTER)
3894 024066  002435                  TMP9+1
3895 024070  000205                  RTS     R5
3896
3897                          ;-------------------------------------------------------------------------
3898                          ; GETT2 -- GET THE "T2" FLAG FROM THE VIA'S IFR REGISTER AND PUT IT
3899                          ;          INTO THE "CARRY" BIT
3900                          ;
3901                          ;-------------------------------------------------------------------------
3902
3903 024072  004537  003616  GETT2:  JSR     R5,READ          ;GET VIA'S IFR REG.
3904 024076  120015                  IFR
3905 024100  002444                  TMPD
3906 024102  103003                  BCC     1$               ;IF NO ERROR, PROCEED
3907 024104                          ERROR                    ;ELSE, REPORT IT
     024104  104460                                                        TRAP    C$ERROR
3908 024106  000262                  SEV                      ;FLAG AN ERROR TO MAINLINE ROUTINE
3909 024110  000207                  RTS     PC               ;  AND TAKE AN ABNORMAL RETURN
3910
3911 024112  010046          1$:     MOV     R0,-(SP)         ;PRESERVE R0
3912 024114  113700  002444          MOVB    TMPD,R0          ;PUT VALUE HERE TO PRESERVE TMP0
3913 024120  106100                  ROLB    R0               ;"IRQ" GOES INTO CARRY BIT
3914 024122  106100                  ROLB    R0               ;"T1" GOES INTO CARRY BIT
3915 024124  106100                  ROLB    R0               ;"T2" GOES INTO CARRY BIT
3916 024126  012600                  MOV     (SP)+,R0         ;RESTORE R0
3917 024130  000207                  RTS     PC
3918
```

```
3930                                    .SBTTL TEST 2 -- VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE

                                        ;+*********************************************************************
                                        ;*
                                        ;*      TEST 2 -- VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE
                                        ;*
                                        ;* A MASTER CLEAR IS DONE, THEN THE SHIFT REG IS PLACED IN INPUT MODE
                                        ;* UNDER CONTROL OF VIA CLK, BY SETTING ACR BIT 4 TO 0, BIT 3 TO 1, AND BIT 2
                                        ;* TO 0. THE PROGRAM CHECKS FOR THE SR FLAG (BIT 2) IN THE IFR TO BE INITIALLY
                                        ;* CLEARED. THEN, THE SR IS LOADED TO INITIALIZE THE SR OPERATION, AND THE
                                        ;* PROGRAM CHECKS FOR SR FLAG = 1 AFTER ABOUT 8 US, AND READS SR REGISTER TO
                                        ;* VERIFY THAT SHIFTING OCCURRED.
                                        ;*
                                        ;-*********************************************************************

                                        ;       BGNTST
     024132                                                                                     T2::
3931 024132  004737  003514             ;       JSR     PC,MSTCLR       ;INIT DMV & ENTER M-LOOP
3932 024136  103003                     ;       BCC     1$              ;IF NO ERROR, PROCEED WITH TESTING
3933 024140                             ;       ERROR                   ;ELSE, REPORT ERROR
     024140  104460                                                                             TRAP    C$ERROR
3934 024142                             ;       ESCAPE  TST             ;   & EXIT TEST
     024142  104410                                                                             TRAP    C$ESCAPE
     024144  000612                                                                            .WORD   L10031-.
3935 024146  005037  002437             1$:     CLR     TMPA+1          ;CLEAR THE "WRITE" DATA FOR ERROR MESSAGES
3936 024152  005037  002252             ;       CLR     TDATA           ;THIS IS A FLAG TO INDICATE THAT "SR" HASN'T
3937                                    ;                               ;BEEN LOADED YET.
3938 024156  004537  003616             ;       JSR     R5,READ         ;GET CURRENT "ACR" CONTENTS (SHOULD BE 000)
3939 024162  120013                     ;       ACR
3940 024164  002440                     ;       TMPB
3941 024166  103003                     ;       BCC     .+10            ;IF NO ERROR, PROCEED
3942 024170                             ;       ERROR                   ;ELSE, REPORT IT
     024170  104460                                                                             TRAP    C$ERROR
3943 024172                             ;       ESCAPE  TST             ;       AND EXIT THIS TEST
     024172  104410                                                                             TRAP    C$ESCAPE
     024174  000562                                                                            .WORD   L10031-.
3944 024176  113737  002440  002441     ;       MOVB    TMPB,TMPB+1     ;MOVE IT FROM I/P BUFFER TO O/P BUFFER
3945 024204  142737  000034  002441     ;       BICB    #<BIT2+BIT3+BIT4>,TMPB+1  ;MAKE SURE CURRENT MODE IS 0
3946 024212  004537  004042             ;       JSR     R5,WRITE        ;FORCE IT TO THAT MODE (MODE 0)
3947 024216  120013                     ;       ACR
3948 024220  002441                     ;       TMPB+1
3949 024222  103003                     ;       BCC     .+10            ;IF NO ERROR, PROCEED
3950 024224                             ;       ERROR                   ;ELSE, REPORT IT
     024224  104460                                                                             TRAP    C$ERROR
3951 024226                             ;       ESCAPE  TST             ;       AND EXIT THIS TEST
     024226  104410                                                                             TRAP    C$ESCAPE
     024230  000526                                                                            .WORD   L10031-.
3952 024232  004537  003616             ;       JSR     R5,READ         ;READ IER INCASE IT'S NEEDED FOR ERROR MESSAGES
3953 024236  120016                     ;       IENR
3954 024240  002446                     ;       TMPE
3955 024242  103003                     ;       BCC     .+10            ;IF NO ERROR, PROCEED
3956 024244                             ;       ERROR                   ;ELSE, REPORT IT
     024244  104460                                                                             TRAP    C$ERROR
3957 024246                             ;       ESCAPE  TST             ;       AND EXIT THIS TEST
     024246  104410                                                                             TRAP    C$ESCAPE
     024250  000506                                                                            .WORD   L10031-.
3958 024252  004737  024760             ;       JSR     PC,GETSR        ;SAMPLE SR INTERRUPT FLAG - IT SHOULD BE 0
```

```
3959 024256  102002                        BVC     .+6             ;IF NO ERROR, PROCEED
3960 024260                                ESCAPE  TST             ;ELSE, IT'S ALREADY BEEN REPORTED - EXIT
     024260  104410                                                          TRAP    C$ESCAPE
     024262  000474                                                          .WORD   L10031-.
3961 024264  103014                        BCC     4$              ;IT IS, GOOD.
3962 024266  004537  003616                JSR     R5,READ         ;READ SR FOR ERROR MESSAGE
3963 024272  120012                        SR
3964 024274  002436                        TMPA
3965 024276  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
3966 024300                                ERROR                   ;ELSE, REPORT IT
     024300  104460                                                          TRAP    C$ERROR
3967 024302                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024302  104410                                                          TRAP    C$ESCAPE
     024304  000452                                                          .WORD   L10031-.
3968 024306                                GEDF    EM52A,ERR52     ;IT ISN'T!  REPORT SR NOT INITIALLY CLEARED
                                                                   ;       "DEVICE FATAL" ERROR # 21
     024306  104455                                                          TRAP    C$ERDF
     024310  000025                                                          .WORD   21
     024312  021457                                                          .WORD   EM52A
     024314  011270                                                          .WORD   ERR52
3969 024316  152737  000010  002441  4$:     BISB    #BIT3,TMPB+1    ;SET SHIFT REG. TO MODE 2
3970 024324  004537  004042                JSR     R5,WRITE
3971 024330  120013                        ACR
3972 024332  002441                        TMPB+1
3973 024334  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
3974 024336                                ERROR                   ;ELSE, REPORT IT
     024336  104460                                                          TRAP    C$ERROR
3975 024340                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024340  104410                                                          TRAP    C$ESCAPE
     024342  000414                                                          .WORD   L10031-.
3976 024344  112737  000204  002447        MOVB    #BIT7+BIT2,TMPE+1  ;ENABLE SR INTERRUPTS WITHIN DMV-11
3977 024352  004537  004042                JSR     R5,WRITE        ;  (WE WILL NOT BE ALLOWING THEM TO GIVE US
3978 024356  120016                        IENR                    ;   A Q-BUS INTERRUPT)
3979 024360  002447                        TMPE+1
3980 024362  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
3981 024364                                ERROR                   ;ELSE, REPORT IT
     024364  104460                                                          TRAP    C$ERROR
3982 024366                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024366  104410                                                          TRAP    C$ESCAPE
     024370  000366                                                          .WORD   L10031-.
3983 024372  004537  003616                JSR     R5,READ         ;READ IER INCASE IT'S NEEDED FOR ERROR MESSAGES
3984 024376  120016                        IENR
3985 024400  002446                        TMPE
3986 024402  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
3987 024404                                ERROR                   ;ELSE, REPORT IT
     024404  104460                                                          TRAP    C$ERROR
3988 024406                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024406  104410                                                          TRAP    C$ESCAPE
     024410  000346                                                          .WORD   L10031-.
3989 024412  105037  002437                CLRB    TMPA+1          ;LOAD SR WITH PROPER VALUE....
3990 024416  005737  002366                TST     BRDTYP          ;  NOTE: THE INPUT LEAD (CB2) WILL EITHER BE
3991 024422  001403                        BEQ     5$              ;        TIED HI(M8064) OR LO(M8053).
3992 024424  112737  000377  002437        MOVB    #377,TMPA+1     ; IF M8064, THEN LOAD SR WITH 000.
3993 024432  004537  004042          5$:   JSR     R5,WRITE        ; IF M8053, THEN LOAD SR WITH 377.
3994 024436  120012                        SR                      ;  THIS ALSO STARTS THE SHIFTING OPERATION.
3995 024440  002437                        TMPA+1                  ;
3996
```

TEST 2 -- VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE

```
3997 024442  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
3998 024444                                  ERROR                   ;ELSE, REPORT IT
     024444  104460                                                                  TRAP    C$ERROR
3999 024446                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     024446  104410                                                                  TRAP    C$ESCAPE
     024450  000306                                                                  .WORD   L10031-.
4000
4001 024452  005337  002252                  DEC     TDATA           ;INDICATE THAT "SR" HAS BEEN LOADED NOW
4002 024456  012703  000100                  MOV     #100,R3         ;GIVE THE INTERRUPT A CHANCE TO HAPPEN
4003 024462  077301                          SOB     R3,.
4004 024464  132777  000004  155634          BITB    #BIT2,@#SEL3    ;DID AN SR INTERRUPT OCCUR WITHIN THE 6502?
4005 024472  001026                          BNE     6$              ;YES, GOOD.
4006 024474  004537  003616                  JSR     R5,READ         ;NO,  SETUP TO REPORT THE ERROR:
4007 024500  120015                          IFR                     ;  GET INTERRUPT FLAG REGISTER
4008 024502  002444                          TMPD
4009 024504  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4010 024506                                  ERROR                   ;ELSE, REPORT IT
     024506  104460                                                                  TRAP    C$ERROR
4011 024510                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     024510  104410                                                                  TRAP    C$ESCAPE
     024512  000244                                                                  .WORD   L10031-.
4012 024514  004537  003616                  JSR     R5,READ         ;  GET FINAL SR CONTENTS -- SHOULD BE 0
4013 024520  120012                          SR
4014 024522  002436                          TMPA
4015 024524  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4016 024526                                  ERROR                   ;ELSE, REPORT IT
     024526  104460                                                                  TRAP    C$ERROR
4017 024530                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     024530  104410                                                                  TRAP    C$ESCAPE
     024532  000224                                                                  .WORD   L10031-.
4018 024534                                  GEDF    EM52B,ERR52     ;REPORT MISSING SR INTERRUPT WITHIN DMV-11
                                                                     ;       "DEVICE FATAL" ERROR # 22
     024534  104455                                                                  TRAP    C$ERDF
     024536  000026                                                                  .WORD   22
     024540  021535                                                                  .WORD   EM52B
     024542  011270                                                                  .WORD   ERR52
4019 024544                                  ESCAPE  TST             ;FURTHER TESTING INVALID
     024544  104410                                                                  TRAP    C$ESCAPE
     024546  000210                                                                  .WORD   L10031-.
4020 024550  004537  003616          6$:     JSR     R5,READ         ;GET FINAL SR CONTENTS:
4021 024554  120012                          SR                      ; IF M8064, THEN SR SHOULD=377
4022 024556  002436                          TMPA                    ; IF M8053, THEN SR SHOULD=000
4023 024560  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4024 024562                                  ERROR                   ;ELSE, REPORT IT
     024562  104460                                                                  TRAP    C$ERROR
4025 024564                                  ESCAPE  TST             ;       AND EXIT THIS TEST
     024564  104410                                                                  TRAP    C$ESCAPE
     024566  000170                                                                  .WORD   L10031-.
4026
4027 024570  005737  002366                  TST     BRDTYP          ;CHECK DMV-11 BOARD TYPE
4028 024574  001005                          BNE     9$              ;
4029 024576  122737  000377  002436          CMPB    #377,TMPA       ;M8064::SEE IF CORRECT RESULT
4030 024604  001422                          BEQ     8$              ;       YES:GOOD
4031 024606  000403                          BR      7$              ;       NO: GO REPORT ERROR
4032 024610  105737  002436          9$:     TSTB    TMPA            ;M8053::SEE IF CORRECT RESULT
4033 024614  001416                          BEQ     8$              ;       YES:GOOD.
4034 024616  004537  003616          7$:     JSR     R5,READ         ; NO: SETUP TO REPORT THE ERROR:
```

TEST 2 -- VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE

```
4035 024622  120015                        IFR                     ;  GET INTERRUPT FLAG REGISTER
4036 024624  002444                        TMPD
4037 024626  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
4038 024630                                ERROR                   ;ELSE, REPORT IT
     024630  104460                                                                TRAP    C$ERROR
4039 024632                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024632  104410                                                                TRAP    C$ESCAPE
     024634  000122                                                                .WORD   L10031-.
4040 024636                                GEDF    EM52C,ERR52     ;REPORT INCOMPLETE OR BAD SHIFTING OPERATION
                                                                   ;       "DEVICE FATAL" ERROR # 23
     024636  104455                                                                TRAP    C$ERDF
     024640  000027                                                                .WORD   23
     024642  021567                                                                .WORD   EM52C
     024644  011270                                                                .WORD   ERR52
4041 024646                                ESCAPE  TST             ;FURTHER TESTING INVALID
     024646  104410                                                                TRAP    C$ESCAPE
     024650  000106                                                                .WORD   L10031-.
4042 024652  105077  155450          8$:     CLRB    @#BSEL3         ;CLEAR THE INTERRUPT FLAGS
4043 024656  004537  003616                  JSR     R5,READ         ;HIT THE SHIFT REG. THIS TIME WITH A READ
4044 024662  120012                        SR                      ; (WE DON'T REALLY CARE THIS TIME WHAT THE DATA
4045 024664  002436                        TMPA                    ;  RETURNED IS. BUT, WE HAVE TO PUT IT SOMEWHERE
4046 024666  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
4047 024670                                ERROR                   ;ELSE, REPORT IT
     024670  104460                                                                TRAP    C$ERROR
4048 024672                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024672  104410                                                                TRAP    C$ESCAPE
     024674  000062                                                                .WORD   L10031-.
4049 024676  004737  005132                JSR     PC,STALL        ;DELAY FOR A LITTLE WHILE TO LET THE INTERRUPT
4050 024702  004737  005132                JSR     PC,STALL        ;  GET THROUGH
4051 024706  004737  005132                JSR     PC,STALL
4052 024712  132777  000004  155406        BITB    #BIT2,@#BSEL3   ;DID WE GET AN INTERRUPT ON THE READ OPERATION?
4053 024720  001016                        BNE     10$             ;YES, GOOD.
4054 024722  004537  003616                JSR     R5,READ         ;NO,  SETUP TO REPORT THE ERROR:
4055 024726  120015                        IFR                     ;  GET INTERRUPT FLAG REGISTER
4056 024730  002444                        TMPD
4057 024732  103003                        BCC     .+10            ;IF NO ERROR, PROCEED
4058 024734                                ERROR                   ;ELSE, REPORT IT
     024734  104460                                                                TRAP    C$ERROR
4059 024736                                ESCAPE  TST             ;       AND EXIT THIS TEST
     024736  104410                                                                TRAP    C$ESCAPE
     024740  000016                                                                .WORD   L10031-.
4060 024742                                GEDF    EM52D,ERR52     ;REPORT THE FAILURE.
                                                                   ;       "DEVICE FATAL" ERROR # 24
     024742  104455                                                                TRAP    C$ERDF
     024744  000030                                                                .WORD   24
     024746  021653                                                                .WORD   EM52D
     024750  011270                                                                .WORD   ERR52
4061 024752                                ESCAPE  TST
     024752  104410                                                                TRAP    C$ESCAPE
     024754  000002                                                                .WORD   L10031-.
4062 024756                          10$:
4063 024756                                ENDTST
     024756                                                             L10031:
     024756  104401                                                                TRAP    C$ETST
4064 024760  004537  003616          GETSR:  JSR     R5,READ
4065 024764  120015                        IFR                     ;GET CURRENT INTERRUPT FLAG REGISTER SETTINGS
4066 024766  002444                        TMPD
```

TEST 2 -- VIA'S SR INPUT (MODE 2) - SYSTEM CLOCK MODE

```
4067 024770  103003                    BCC     1$              ;IF NO ERROR, PROCEED
4068 024772                            ERROR                   ;ELSE, REPORT IT
     024772  104460                                                    TRAP    C$ERROR
4069 024774  000262                    SEV                     ;FLAG AN ERROR TO MAINLINE ROUTINE
4070 024776  000207                    RTS     PC              ;  AND TAKE AN ABNORMAL RETURN
4071
4072 025000  010046                1$: MOV     R0,-(SP)        ;SAVE REGISTER FOR CALLER
4073 025002  113700  002444            MOVB    TMPD,R0         ;PUT THEM WHERE WE CAN EASILY MASAGE THEM
4074 025006  106000                    RORB    R0              ;CA2 ---> CARRY BIT
4075 025010  106000                    RORB    R0              ;CA1 ---> CARRY BIT
4076 025012  106000                    RORB    R0              ;SR ---> CARRY BIT
4077 025014  012600                    MOV     (SP)+,R0        ;RESTORE REGISTER
4078 025016  000207                    RTS     PC              ;RETURN WITH SR INTERRUPT FLAG IN CARRY BIT
4079
```

```
                        .SBTTL TEST 3 -- NPR CONTROL REGISTER - MASTER CLEAR

                        ;+*********************************************************************
                        ;*
                        ;*      TEST 3 -- NPR CONTROL REGISTER - MASTER CLEAR
                        ;*
                        ;* THE PROGRAM SETS THE FOLLOWING BITS IN THE NPR CONTROL REGISTER :
                        ;* IN/OUT, BYTE OPER, AND DISABL INIT. THE REGISTER IS READ AND VERIFIED.
                        ;* THEN, A MASTER CLEAR IS PERFORMED, AND THE REGISTER IS READ AND CHECKED FOR
                        ;* 000.
                        ;*
                        ;--*********************************************************************
                        ;
                        ;       BGNTST
     025020                                                                  T3::
4090 025020  004737  003514          JSR     PC,MSTCLR       ;INIT DMV & START UP MAINT. LOOP
4091 025024  103003                  BCC     1$              ;IF NO ERROR, PROCEED WITH TESTING
4092 025026                          ERROR                   ;ELSE REPORT ERROR
     025026  104460                                                      TRAP    C$ERROR
4093 025030                          ESCAPE  TST             ;   & EXIT TEST
     025030  104410                                                      TRAP    C$ESCAPE
     025032  000352                                                      .WORD   L10032-.
4094
4095 025034  004737  005134    1$:   JSR     PC,NPREAD       ;GET CONTENTS OF ALL NPR REGISTERS INTO BT2
4096 025040  103002                  BCC     30$             ;IF AN ERROR OCCURED,
4097 025042                          ERROR                   ;REPORT IT &
     025042  104460                                                      TRAP    C$ERROR
4098 025044  000557                  BR      24$             ;  EXIT
4099
4100 025046  012702  002740    30$:  MOV     #BT2,R2         ;POINT TO NPR REGISTER CONTENTS
4101 025052  010237  002256          MOV     R2,BDATA        ;USE IT ALSO FOR ERROR HANDLING
4102 025056  010237  025172          MOV     R2,13$          ;SETUP ALSO FOR READ BACK
4103
4104 025062  004537  005004          JSR     R5,MOVSW        ;GET THE "EXPECTED" RESULTS TOO
4105 025066  002632                  NPRMCR
4106 025070  002654    12$:  BT1
4107 025072  000011    11$:  9.
4108 025074  013701  025070          MOV     12$,R1          ;POINT TO TABLE OF EXPECTED REGISTER CONTENTS
4109 025100  010137  002254          MOV     R1,GDATA        ;USE IT ALSO FOR ERROR HANDLING
4110 025104  012703  000001          MOV     #1,R3           ;COUNT OF # OF NPR REGISTERS BEING PROCESSED
4111                                                         ;FOR NOW,  ONLY THE CONTROL REGISTER IS CHECKED!!
4112
4113                                 ;*************************************************************
4114                                 ; PLEASE NOTE THAT "GDATA" & "BDATA" NOW CONTAIN POINTERS -- NOT DATA!
4115                                 ; THIS IS A DEVIANT AND THEREFORE SHOULD BE BORNE IN MIND WHEN TRYING TO
4116                                 ; FOLLOW THIS DEVIOUS LOGIC.
4117                                 ;*************************************************************
4118
4119 025110  121112    2$:   CMPB    (R1),(R2)       ;CHECK ONE BYTE
4120 025112  001003          BNE     3$              ;GO REPORT FAILURE IF ANY ERROR IS FOUND
4121 025114  022122          CMP     (R1)+,(R2)+     ;BUMP POINTERS -- TABLES ARE ACTUALLY WORD TABLES
4122 025116  077304          SOB     R3,2$           ;LOOP IF NOT DONE YET
4123 025120  000412          BR      4$              ;ELSE, PROCEED WITH TESTING
4124
4125 025122  163701  025070    3$:   SUB     12$,R1          ;CALCULATE THE REGISTER # CAUSING THE FAILURE
4126 025126  010137  002300          MOV     R1,REGNUM       ;IDENTIFY FAULTY REGISTER
4127 025132                          GEDF    EM26,ERR8       ;NPR ERROR - BAD INITIALIZATION
```

```
                                                        ;       "DEVICE FATAL" ERROR # 25
     025132  104455                                                              TRAP    C$ERDF
     025134  000031                                                              .WORD   25
     025136  016152                                                              .WORD   EM26
     025140  006532                                                              .WORD   ERR8
4128 025142                          ESCAPE  TST             ;IF IT CAN'T BE INIT'ED, WE CAN'T TEST IT!
     025142  104410                                                              TRAP    C$ESCAPE
     025144  000240                                                              .WORD   L10032-.
4129 025146  004537  004054     4$:  JSR     R5,WRITEI       ;WRITE THE TEST PATTERN INTO THE CONTROL REG.
4130 025152  123004                 NPRCTL
4131 025154  000355             5$:  355                     ;THIS SET SEVERAL BITS -- BUT NOT ALL!
4132 025156  103002                 BCC     31$             ;IF AN ERROR OCCURED,
4133 025160                          ERROR                   ;REPORT IT &
     025160  104460                                                              TRAP    C$ERROR
4134 025162  000510                 BR      24$             ;  EXIT
4135 025164  004537  003616     31$: JSR     R5,READ         ;READ IT BACK
4136 025170  123004                 NPRCTL
4137 025172  002740             13$: BIT2
4138 025174  103002                 BCC     32$             ;IF AN ERROR OCCURED,
4139 025176                          ERROR                   ;REPORT IT &
     025176  104460                                                              TRAP    C$ERROR
4140 025200  000501                 BR      24$             ;  EXIT
4141
4142 025202  013777  025154  155044 32$: MOV     5$,@GDATA       ;BUILD THE EXPECTED RETURN VALUE
4143 025210  042777  000300  155036      BIC     #NPRABT!NPRGO,@GDATA    ;WE CAN'T WRITE THESE BITS
4144
4145 025216  004737  005134             JSR     PC,NPREAD       ;GET CONTENTS OF ALL NPR REGISTERS INTO BIT2
4146 025222  103002                     BCC     33$             ;IF AN ERROR OCCURED,
4147 025224                              ERROR                   ;REPORT IT &
     025224  104460                                                              TRAP    C$ERROR
4148 025226  000466                     BR      24$             ;  EXIT
4149 025230  013701  002254     33$:     MOV     GDATA,R1        ;POINT TO TABLE OF EXPECTED REGISTER CONTENTS
4150 025234  013702  002256             MOV     BDATA,R2        ;POINT TO NPR REGISTER CONTENTS
4151 025240  012703  000001             MOV     #1,R3           ;COUNT OF # OF NPR REGISTERS BEING PROCESSED
4152                                                            ;FOR NOW,  ONLY THE CONTROL REGISTER IS CHECKED!!
4153
4154 025244  121112             7$:      CMPB    (R1),(R2)       ;CHECK ONE BYTE
4155 025246  001003                     BNE     8$              ;GO REPORT FAILURE IF ANY ERROR IS FOUND
4156 025250  022122                     CMP     (R1)+,(R2)+     ;BUMP POINTERS -- TABLES ARE ACTUALLY WORD TABLES
4157 025252  077304                     SOB     R3,7$           ;LOOP IF NOT DONE YET
4158 025254  000412                     BR      9$              ;ELSE, PROCEED WITH TESTING
4159
4160 025256  163701  025070     8$:      SUB     12$,R1          ;CALCULATE THE REGISTER # CAUSING THE FAILURE
4161 025262  010137  002300             MOV     R1,REGNUM       ;IDENTIFY FAULTY REGISTER
4162 025266                              CEDF    EM26A,ERR8      ;NPR ERROR -- DURING WRITE TO CONTROL REG.
                                                        ;       "DEVICE FATAL" ERROR # 26
     025266  104455                                                              TRAP    C$ERDF
     025270  000032                                                              .WORD   26
     025272  016204                                                              .WORD   EM26A
     025274  006532                                                              .WORD   ERR8
4163 025276                          ESCAPE  TST             ;IF IT CAN'T BE INIT'ED, WE CAN'T TEST IT!
     025276  104410                                                              TRAP    C$ESCAPE
     025300  000104                                                              .WORD   L10032-.
4164 025302  004737  003514     9$:  JSR     PC,MSTCLR       ;RE-INITIALIZE DMV & ENTER M-LOOP
4165 025306  103003                 BCC     20$             ;IF NO ERROR, PROCEED WITH TESTING
4166 025310                          ERROR                   ;ELSE REPORT IT
     025310  104460                                                              TRAP    C$ERROR
```

TEST 3 -- NPR CONTROL REGISTER - MASTER CLEAR

```
4167 025312                                    ESCAPE  TST             ;   & EXIT TEST
     025312  104410                                                                 TRAP    C$ESCAPE
     025314  000070                                                                 .WORD   L10032-.
4168
4169                                   ;       THE "MASTER CLEAR" JUST PERFORMED SHOULD RESET THE NPR CONTROL
4170                                   ;       REGISTER.  IT SHOULD NOW EQUAL 004 AGAIN.
4171
4172 025316  013777  002632  154730  20$:    MOV     NPRMCR,@GDATA   ;RESET THE EXPECTED DATA
4173
4174                                   ;       ALSO, THE OTHER REGISTERS SHOULD STILL BE AT THEIR INITIAL VALUES
4175
4176 025324  004737  005134                    JSR     PC,NPREAD       ;GET CONTENTS OF ALL NPR REGISTERS INTO BT2
4177 025330  103002                            BCC     34$             ;IF AN ERROR OCCURED,
4178 025332                                    ERROR                   ;REPORT IT &
     025332  104460                                                                 TRAP    C$ERROR
4179 025334  000423                            BR      24$             ;  EXIT
4180 025336  013701  002254            34$:    MOV     GDATA,R1        ;POINT TO TABLE OF EXPECTED REGISTER CONTENTS
4181 025342  013702  002256                    MOV     BDATA,R2        ;POINT TO NPR REGISTER CONTENTS
4182 025346  012703  000001                    MOV     #1,R3           ;COUNT OF # OF NPR REGISTERS BEING PROCESSED
4183                                                                   ;FOR NOW,  ONLY THE CONTROL REGISTER IS CHECKED!!
4184
4185 025352  121112            21$:    CMPB    (R1),(R2)       ;CHECK ONE BYTE
4186 025354  001003                    BNE     22$             ;GO REPORT FAILURE IF ANY ERROR IS FOUND
4187 025356  022122                    CMP     (R1)+,(R2)+     ;BUMP POINTERS -- TABLES ARE ACTUALLY WORD TABLES
4188 025360  077304                    SOB     R3,21$          ;LOOP IF NOT DONE YET
4189 025362  000410                    BR      24$             ;ELSE, PROCEED WITH TESTING
4190
4191 025364  163701  025070    22$:    SUB     12$,R1          ;CALCULATE THE REGISTER # CAUSING THE FAILURE
4192 025370  010137  002300            MOV     R1,REGNUM       ;IDENTIFY FAULTY REGISTER
4193 025374                            GEDF    EM26,ERR8       ;NPR ERROR -- BAD INITIALIZATION
                                       ;               "DEVICE FATAL" ERROR # 27
     025374  104455                                                                 TRAP    C$ERDF
     025376  000033                                                                 .WORD   27
     025400  016152                                                                 .WORD   EM26
     025402  006532                                                                 .WORD   ERR8
4194 025404                    24$:    ENDTST
     025404                                                                 L10032:
     025404  104401                                                                 TRAP    C$ETST
```

```
                        .SBTTL TEST 4 -- NPR DATA-OUT

                        ;+*********************************************************************
                        ;*
                        ;*      TEST 4 -- NPR DATA-OUT
                        ;*
                        ;* FIRST SUBTEST :
                        ;* THE NPR OUTPUT ADDRESS REGISTER IS LOADED WITH THE ADDRESS OF A 2 BYTE
                        ;* BUFFER IN THE PROGRAM. THEN, EACH WORD OF DATA PATTERN F IS LOADED INTO THE
                        ;* NPR OUTPUT DATA REGISTER, A FULLWORD NPR OUTPUT REQUEST IS PERFORMED,
                        ;* AND THE PROGRAM CHECKS FOR THE CORRECT DATA IN THE PROGRAM BUFFER. ALSO,
                        ;* THE PROGRAM CHECKS THAT THE ABORT XFER BIT IN THE NPR CONTROL REGISTER
                        ;* NEVER GETS SET.
                        ;*   DATA PATTERN F = 125252, 052525, 000000, 177777, 000001, 000002, 000004,
                        ;*                    000010, 000020, 000040, 000100, 000200, 000400, 001000,
                        ;*                    002000, 004000, 010000, 020000, 040000, 100000, 177776,
                        ;*                    177775, 177773, 177767, 177757, 177737, 177677, 177577,
                        ;*                    177377, 176777, 175777, 173777, 167777, 157777, 137777,
                        ;*                    077777, 000000
                        ;*
                        ;* SECOND SUBTEST:
                        ;* THE ABOVE OPERATIONS ARE REPEATED IN BYTE NPR TRANSFER MODE, USING THE DATA
                        ;* BYTES IN DATA PATTERN B. THE LOW BYTE OF THE PROGRAM BUFFER IS USED, AND
                        ;* THE UPPER BYTE IS CLEARED AT THE START, AND IS CHECKED TO REMAIN UNCHANGED
                        ;* THROUGHOUT THE SUBTEST.
                        ;*   DATA PATTERN B = 125, 252, 000, 377, 001, 002, 004, 010, 020, 040, 100,
                        ;*                    200, 376, 375, 373, 367, 357, 337, 277, 177, 000
                        ;*
                        ;*
                        ;-*********************************************************************
                        ;
                        ;       BGNTST                                                  T4::
     025406
4223                    000004                  NPROTS = $T
4224 025406  004737     003514                  JSR     PC,MSTCLR       ;INIT DMV & START UP MAINT. LOOP
4225 025412  103003                             BCC     1$              ;IF NO ERROR, PROCEED WITH TEST
4226 025414                                     ERROR                   ;ELSE, REPORT IT
     025414  104460                                                                     TRAP    C$ERROR
4227 025416                                     ESCAPE  TST             ;   & EXIT TEST
     025416  104410                                                                     TRAP    C$ESCAPE
     025420  000474                                                                     .WORD   L10033-.
4228
4229                                    ;---------- ---------- ----------
4230
4231 025422                             1$:     BGNSUB                  ;----- MAIN MEMORY WORD DATA-OUT TESTING ----
     025422                                                                     T4.1:
     025422  104402                                                                     TRAP    C$BSUB
4232 025424  013737     002506  025442         MOV     PATF,4$         ;SETUP COUNT OF # OF WORDS IN TEST PATTERN
4233
4234 025432  004537     005266                 JSR     R5,NPRMOV       ;MOVE DATA THROUGH THE NPR LOGIC
4235 025436  002510                     2$:     PATF+2                  ;  ADDRESS OF DATA
4236 025440  002654                     3$:     BUFAREA                 ;  BUFFER AREA
4237 025442  000000                     4$:     0                       ;*** MODIFIED FROM ABOVE *** -- WORD COUNT
4238 025444  000044                             NPRDL                   ;  OPERATION TO BE UTILIZED
4239
4240 025446  103025                             BCC     7$              ;IF ERROR, REPORT IT
4241 025450                             13$:    ERROR
```

TEST 4 -- NPR DATA-OUT

```
                                                                                    TRAP    C$ERROR
4242 025452  005737  002412                TST     TMPO            ;WE JUST REPORTED ONE ERROR BUT WAS IT A TIMEOUT
4243 025456  001421                        BEQ     7$              ;ERROR? IF SO, PROCEED WITH TESTING.  ELSE,
4244 025460  022737  000006  002204        CMP     #NPRTOE,ERRNBR  ;WE WILL HAVE TO REPORT IT HERE AND NOW.
4245 025466  001415                        BEQ     7$              ;THE TIMEOUT ERROR WAS ALREADY REPORTED.
4246 025470  012737  000002  002202        MOV     #T.EHRD,ERRTYP  ;IT WASN'T REPORTED YET, SETUP FOR IT NOW:
4247 025476  012737  000006  002204        MOV     #NPRTOE,ERRNBR
4248 025504  012737  016316  002206        MOV     #EM26E,ERRMSG
4249 025512  012737  007314  002210        MOV     #ERR11,ERRBLK
4250 025520  000753                        BR      13$             ;LOOP BACK TO CAUSE REPORT @ PROPER PC LOCATION
4251
4252 025522  013701  025436          7$:   MOV     2$,R1           ;POINT TO GOOD DATA
4253 025526  013702  025440                MOV     3$,R2           ;       & ACTUAL DATA
4254 025532  013703  025442                MOV     4$,R3           ;GET WORD COUNT
4255 025536  005037  002276                CLR     ERRFLG          ;RESET ERROR FLAG
4256
4257 025542  022122                  5$:   CMP     (R1)+,(R2)+     ;CHECK RECEIVED DATA
4258 025544  001007                        BNE     6$              ;ERROR, GO REPORT IT
4259 025546  077303                 11$:   SOB     R3,5$           ;GOOD,  IF MORE DO IT AGAIN
4260 025550  005737  002276                TST     ERRFLG          ;ELSE, SEE IF WE MUST FINISH AN ERROR MESSAGE
4261 025554  001440                        BEQ     10$             ;NO,  TEST IT AGAIN BUT WITH BYTE TRANSFERS
4262 025556  004737  013076                JSR     PC,NULERR       ;YES, USE COMMON ROUTINE TO END ERROR MESSAGE
4263 025562  000435                        BR      10$             ;WE CAN TEST IT AGAIN BUT WITH BYTE TRANSFERS
4264
4265 025564  010146                  6$:   MOV     R1,-(SP)        ;SAVE THIS FOR FURTHER TESTING
4266 025566  014137  002254                MOV     -(R1),GDATA     ;SETUP FOR ERROR REPORT
4267 025572  014237  002256                MOV     -(R2),BDATA
4268 025576  010237  002252                MOV     R2,TDATA        ;LSI-11'S MEMORY ADDRESS
4269 025602  163701  025436                SUB     2$,R1           ;CALCULATE THE OFFSET AT WHICH THE
4270 025606  006201                        ASR     R1              ;  DATA COMPARISON ERROR OCCURED
4271 025610  010137  002300                MOV     R1,REGNUM       ;THE ERROR MESSAGE WILL REPORT THIS TOO
4272 025614  005737  002276                TST     ERRFLG          ;HAVE WE ALREADY REPORTED AN ERROR HERE?
4273 025620  001007                        BNE     8$              ;YES, THEN WE ONLY PRINT DATA THIS TIME
4274 025622  005237  002276                INC     ERRFLG          ;NO,  SET FLAG & REPORT THE WHOLE MESSAGE
4275 025626                                GEDF    EM26B,ERR9      ;WORD NPR TRANSFER DMV ***> LSI
                                                                   ;       "DEVICE FATAL" ERROR # 28
     025626  104455                                                                 TRAP    C$ERDF
     025630  000034                                                                 .WORD   28
     025632  016226                                                                 .WORD   EM26B
     025634  007300                                                                 .WORD   ERR9
4276 025636  000402                        BR      9$              ;  RESUME TESTING
4277
4278 025640  004737  012574          8$:   JSR     PC,ERR9$        ;IDENTIFY THE FAILING DATA
4279 025644  012601                  9$:   MOV     (SP)+,R1        ;RESTORE POINTERS
4280 025646  013702  002252                MOV     TDATA,R2
4281 025652  005722                        TST     (R2)+
4282 025654  000734                        BR      11$             ;AND RESUME TESTING
4283
4284 025656                          10$:  ENDSUB
     025656                                                                 L10034:
     025656  104403                                                                 TRAP    C$ESUB
4285
4286
4287
4288 025660                                BGNSUB                  ;----- MAIN MEMORY BYTE DATA-OUT TESTING
     025660                                                                 T4.2:
     025660  104402                                                                 TRAP    C$BSUB
```

TEST 4 -- NPR DATA-OUT

```
4289 025662  013737  002456  025700          MOV     PATB,4$          ;SETUP COUNT OF # OF WORDS IN TEST PATTERN
4290
4291 025670  004537  005266                  JSR     R5,NPRMOV        ;MOVE DATA THROUGH THE NPR LOGIC
4292 025674  002460                  2$:     PATB+2                   ;  ADDRESS OF DATA
4293 025676  002654                  3$:     BUFAREA                  ;  BUFFER AREA
4294 025700  000000                  4$:     0                        ;*** MODIFIED FROM ABOVE *** -- BYTE COUNT
4295 025702  000054                          NPRDLB                   ;  OPERATION TO BE UTILIZED
4296
4297 025704  103025                          BCC     7$               ;IF ERROR, REPORT IT
4298 025706                          13$:    ERROR                                                 TRAP    C$ERROR
     025706  104460
4299 025710  005737  002412                  TST     TMP0             ;WE JUST REPORTED ONE ERROR BUT WAS IT A TIMEOUT
4300 025714  001421                          BEQ     7$               ;ERROR? IF SO, PROCEED WITH TESTING.  ELSE,
4301 025716  022737  000006  002204          CMP     #NPRTOE,ERRNBR   ;WE WILL HAVE TO REPORT IT HERE AND NOW.
4302 025724  001415                          BEQ     7$               ;THE TIMEOUT ERROR WAS ALREADY REPORTED.
4303 025726  012737  000002  002202          MOV     #T.EMRD,ERRTYP   ;IT WASN'T REPORTED YET, SETUP FOR IT NOW:
4304 025734  012737  000006  002204          MOV     #NPRTOE,ERRNBR
4305 025742  012737  016316  002206          MOV     #EM26E,ERRMSG
4306 025750  012737  007314  002210          MOV     #ERR11,ERRBLK
4307 025756  000753                          BR      13$              ;LOOP BACK TO CAUSE REPORT @ PROPER PC LOCATION
4308
4309 025760  013701  025674          7$:     MOV     2$,R1            ;POINT TO GOOD DATA
4310 025764  013702  025676                  MOV     3$,R2            ;       & ACTUAL DATA
4311 025770  013703  025700                  MOV     4$,R3            ;GET BYTE COUNT
4312 025774  005037  002276                  CLR     ERRFLG           ;RESET ERROR FLAG
4313
4314 026000  122122                  5$:     CMPB    (R1)+,(R2)+      ;CHECK RECEIVED DATA
4315 026002  001007                          BNE     6$               ;ERROR, GO REPORT IT
4316 026004  077303                  11$:    SOB     R3,5$            ;GOOD.  IF MORE DO IT AGAIN
4317 026006  005737  002276                  TST     ERRFLG           ;ELSE, SEE IF WE MUST FINISH AN ERROR MESSAGE
4318 026012  001437                          BEQ     10$              ;NO,  THEN WE CAN EXIT THE TEST
4319 026014  004737  013076                  JSR     PC,NULERR        ;YES, OUTPUT THE REQUIRED BLANK LINES.  NOW
4320 026020  000434                          BR      10$              ;THEN WE CAN EXIT THE TEST
4321
4322 026022  010146                  6$:     MOV     R1,-(SP)         ;SAVE THIS FOR FURTHER TESTING
4323 026024  114137  002254                  MOVB    -(R1),GDATA      ;SETUP FOR ERROR REPORT
4324 026030  114237  002256                  MOVB    -(R2),BDATA
4325 026034  010237  002252                  MOV     R2,TDATA         ;LSI-11'S MEMORY ADDRESS
4326 026040  163701  025674                  SUB     2$,R1            ;CALCULATE THE OFFSET AT WHICH THE
4327                                                                  ;  DATA COMPARISON ERROR OCCURED
4328 026044  010137  002300                  MOV     R1,REGNUM        ;THE ERROR MESSAGE WILL REPORT THIS TOO
4329 026050  005737  002276                  TST     ERRFLG           ;HAVE WE ALREADY REPORTED AN ERROR HERE?
4330 026054  001007                          BNE     8$               ;YES, THEN WE ONLY PRINT DATA THIS TIME
4331 026056  005237  002276                  INC     ERRFLG           ;NO,  SET FLAG & REPORT THE WHOLE MESSAGE
4332 026062                          GEDF    EM26C,ERR10      ;BYTE NPR TRANSFER DMV ***> LSI
                                                                  ;       "DEVICE FATAL" ERROR # 29
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   29
                                                                                        .WORD   EM26C
                                                                                        .WORD   ERR10
     026062  104455
     026064  000035
     026066  016251
     026070  007306
4333 026072  000402                          BR      9$               ;  RESUME TESTING
4334
4335 026074  004737  012732          8$:     JSR     PC,ERR10$        ;IDENTIFY THE FAILING DATA
4336 026100  012601                  9$:     MOV     (SP)+,R1         ;RESTORE POINTERS
4337 026102  013702  002252                  MOV     TDATA,R2
4338 026106  005202                          INC     R2
4339 026110  000735                          BR      11$              ;AND RESUME TESTING
```

TEST 4 -- NPR DATA-OUT

```
4340
4341 026112                          10$:    ENDSUB
     026112                                                L10035:
     026112  104403                                               TRAP    C$ESUB
4342 026114                                  ENDTST
     026114                                                L10033:
     026114  104401                                               TRAP    C$ETST
```

```
4360                                    .SBTTL TEST 5 -- NPR DATA-IN

                                        ;+*********************************************************************
                                        ;*
                                        ;*      TEST 5 -- NPR DATA-IN
                                        ;*
                                        ;* THE NPR INPUT ADDRESS REGISTER IS LOADED WITH THE ADDRESS OF A 2 BYTE
                                        ;* BUFFER IN THE PROGRAM. THEN, EACH WORD OF DATA PATTERN F IS LOADED INTO THE
                                        ;* PROGRAM BUFFER, A FULLWORD NPR INPUT REQUEST IS ISSUED AND PERFORMED,
                                        ;* AND THE PROGRAM CHECKS FOR THE CORRECT DATA IN THE NPR INPUT DATA REG.
                                        ;* ALSO, THE PROGRAM CHECKS THAT THE ABORT XFER BIT IN THE NPR CONTROL
                                        ;* REGISTER NEVER GETS SET.
                                        ;*   DATA PATTERN F = 125252, 052525, 000000, 177777, 000001, 000002, 000004,
                                        ;*                    000010, 000020, 000040, 000100, 000200, 000400, 001000,
                                        ;*                    002000, 004000, 010000, 020000, 040000, 100000, 177776,
                                        ;*                    177775, 177773, 177767, 177757, 177737, 177677, 177577,
                                        ;*                    177377, 176777, 175777, 173777, 167777, 157777, 137777,
                                        ;*                    077777, 000000
                                        ;*
                                        ;-*********************************************************************
                                        ;
                                        ;       BGNTST
     026116                                                                                  T5::
4361 026116  004737  003514                     JSR     PC,MSTCLR       ;INIT DMV & START UP MAINT. LOOP
4362 026122  103003                             BCC     1$              ;IF NO ERROR, PROCEED WITH TEST
4363 026124                                     ERROR                   ;ELSE, REPORT IT
     026124  104460                                                                          TRAP    C$ERROR
4364 026126                                     ESCAPE  TST             ;   & EXIT TEST
     026126  104410                                                                         TRAP    C$ESCAPE
     026130  000234                                                                         .WORD   L10036-.
4365
4366 026132                             1$:
4367 026132  013737  002506  026150                 MOV     PATF,4$         ;SETUP COUNT OF # OF WORDS IN TEST PATTERN
4368
4369 026140  004537  005266                     JSR     R5,NPRMOV       ;MOVE DATA THROUGH THE NPR LOGIC
4370 026144  002510                     2$:    PATF+2                  ;  ADDRESS OF DATA
4371 026146  002654                     3$:    BUFAREA                 ;  BUFFER AREA
4372 026150  000000                     4$:    0                       ;*** MODIFIED FROM ABOVE *** -- WORD COUNT
4373 026152  000004                            NPRLD                   ;  OPERATION TO BE UTILIZED
4374
4375 026154  103025                             BCC     7$              ;IF ERROR, REPORT IT
4376 026156                             13$:   ERROR
     026156  104460                                                                         TRAP    C$ERROR
4377 026160  005737  002412                     TST     TMP0            ;WE JUST REPORTED ONE ERROR BUT WAS IT A TIMEOUT
4378 026164  001421                             BEQ     7$              ;ERROR? IF SO, PROCEED WITH TESTING.  ELSE,
4379 026166  022737  000006  002204             CMP     #NPRTOE,ERRNBR  ;WE WILL HAVE TO REPORT IT HERE AND NOW.
4380 026174  001415                             BEQ     7$              ;THE TIMEOUT ERROR WAS ALREADY REPORTED.
4381 026176  012737  000002  002202             MOV     #T.EHRD,ERRTYP  ;IT WASN'T REPORTED YET, SETUP FOR IT NOW:
4382 026204  012737  000006  002204             MOV     #NPRTOE,ERRNBR
4383 026212  012737  016316  002206             MOV     #EM26E,ERRMSG
4384 026220  012737  007314  002210             MOV     #ERR11,ERRBLK
4385 026226  000753                             BR      13$             ;LOOP BACK TO CAUSE REPORT @ PROPER PC LOCATION
4386
4387 026230  013701  026144                 7$: MOV     2$,R1           ;POINT TO GOOD DATA
4388 026234  013702  026146                     MOV     3$,R2           ;       & ACTUAL DATA
4389 026240  013703  026150                     MOV     4$,R3           ;GET WORD COUNT
4390 026244  005037  002276                     CLR     ERRFLG          ;RESET ERROR FLAG
```

TEST 5 NPR DATA IN

```
4391
4392 026250  022122                5$:     CMP     (R1)+,(R2)+     ;CHECK RECEIVED DATA
4393 026252  001007                        BNE     6$              ;ERROR, GO REPORT IT
4394 026254  077303                11$:    SOB     R3,5$           ;GOOD,  IF MORE DO IT AGAIN
4395 026256  005737  002276                TST     ERRFLG          ;ELSE, SEE IF WE MUST FINISH AN ERROR MESSAGE
4396 026262  001440                        BEQ     10$             ;NO,  THEN WE CAN EXIT THE TEST
4397 026264  004737  013076                JSR     PC,NULERR       ;YES, USE COMMON ROUTINE TO END ERROR MESSAGE
4398 026270  000435                        BR      10$             ;THEN WE CAN EXIT THE TEST
4399
4400 026272  010246                6$:     MOV     R2,-(SP)        ;SAVE THIS FOR FURTHER TESTING
4401 026274  014137  002254                MOV     (R1),GDATA      ;SETUP FOR ERROR REPORT
4402 026300  014237  002256                MOV     (R2),BDATA
4403 026304  010137  002252                MOV     R1,TDATA        ;LSI-11'S MEMORY ADDRESS
4404 026310  163701  026144                SUB     2$,R1           ;CALCULATE THE OFFSET AT WHICH THE
4405 026314  006201                        ASR     R1              ;  DATA COMPARISON ERROR OCCURED
4406 026316  010137  002300                MOV     R1,REGNUM       ;THE ERROR MESSAGE WILL REPORT THIS TOO
4407 026322  005737  002276                TST     ERRFLG          ;HAVE WE ALREADY REPORTED AN ERROR HERE?
4408 026326  001007                        BNE     8$              ;YES, THEN WE ONLY PRINT DATA THIS TIME
4409
4410 026330  005237  002276                INC     ERRFLG          ;NO,  SET FLAG & REPORT THE WHOLE MESSAGE
4411 026334                                GEDF    EM26D,ERR9      ;WORD NPR TRANSFER LSI ***> DMV
                                                                   ;       "DEVICE FATAL" ERROR # 30
     026334  104455                                                                TRAP    C$ERDF
     026336  000036                                                                .WORD   30
     026340  016274                                                                .WORD   EM26D
     026342  007300                                                                .WORD   ERR9
4412 026344  000402                        BR      9$              ;  RESUME TESTING
4413
4414 026346  004737  012574          8$:   JSR     PC,ERR9$        ;IDENTIFY THE FAILING DATA
4415 026352  012602                9$:     MOV     (SP)+,R2        ;RESTORE POINTERS
4416 026354  013701  002252                MOV     TDATA,R1
4417 026360  005721                        TST     (R1)+
4418 026362  000734                        BR      11$             ;AND RESUME TESTING
4419
4420 026364                        10$:
4421 026364                                ENDTST
     026364                                                                L10036:
     026364  104401                                                                TRAP    C$ETST
```

```
4435                            .SBTTL  TEST 6  - NPR XFER ABORT

                                ;+*********************************************************************
                                ;*
                                ;*      TEST 6  - NPR XFER ABORT
                                ;*
                                ;* FIRST SUBTEST :
                                ;* THE PROGRAM PERFORMS AN OUTPUT NPR REQUEST TO A NON-EXISTENT MEMORY
                                ;* LOCATION, AND CHECKS FOR THE ASSERTION OF ABORT XFER BIT IN THE NPR CONTROL
                                ;* REGISTER. THEN, AN OUTPUT NPR IS DONE AND CHECKED, TO A LOCATION IN THE
                                ;* PROGRAM, USING 125252 FOR DATA, AND THE PROGRAM CHECKS FOR ABORT XFER TO
                                ;* BE CLEARED BY SETTING THE DONE BIT.
                                ;* SECOND SUBTEST :
                                ;* THE ABOVE SUBTEST IS REPEATED USING INPUT NPR'S.
                                ;*
                                ;-*********************************************************************
                                ;
                                ;       BGNTST
                                                                                        T6::
    026366
4436                            ;***  SUBTEST # 1  -  NPR OUTPUT TO NON-EXISTENT LOCATION FORCING NPR-ABORT  ***
4437
4438 026366                     ;       BGNSUB
    026366  104402                                                                      T6.1:
                                                                                        TRAP    C$BSUB
4439 026370  004737  003514            JSR     PC,MSTCLR       ;INIT DMV & ENTER M-LOOP
4440 026374  103003                    BCC     .+10            ;IF NO ERROR, PROCEED WITH TESTING
4441 026376                            ERROR                   ;ELSE, REPORT ERROR
    026376  104460                                                                      TRAP    C$ERROR
4442 026400                            ESCAPE  SUB             ;   & EXIT TEST
    026400  104410                                                                      TRAP    C$ESCAPE
    026402  000164                                                                      .WORD   L10040-.
4443 026404  012737  000001  002412    MOV     #1,TMP0         ;DISABLE PRINTOUT OF TIMEOUT COUNT BY ERR11
4444
4445 026412  012737  160000  002420    MOV     #160000,TMP3    ;SETUP 11'S ADDRESS
4446 026420  012737  000044  002414    MOV     #NPRDL,TMP1     ;CONTROL REG. VALUE FOR NPR-OUT COMMAND
4447 026426  004537  004042            JSR     R5,WRITE        ;SETUP ADDRESS OUT REGISTERS
4448 026432  000070                    NPRAOL
4449 026434  002420                    TMP3
4450 026436  103003                    BCC     .+10            ;IF NO ERROR, PROCEED
4451 026440                            ERROR                   ;ELSE, REPORT IT
    026440  104460                                                                      TRAP    C$ERROR
4452 026442                            ESCAPE  SUB             ;       AND EXIT THIS TEST
    026442  104410                                                                      TRAP    C$ESCAPE
    026444  000122                                                                      .WORD   L10040 .
4453 026446  004537  004042            JSR     R5,WRITE
4454 026452  000071                    NPRAOH
4455 026454  002421                    TMP3+1
4456 026456  103003                    BCC     .+10            ;IF NO ERROR, PROCEED
4457 026460                            ERROR                   ;ELSE, REPORT IT
    026460  104460                                                                      TRAP    C$ERROR
4458 026462                            ESCAPE  SUB             ;       AND EXIT THIS TEST
    026462  104410                                                                      TRAP    C$ESCAPE
    026464  000102                                                                      .WORD   L10040 .
4459 026466  004537  004054            JSR     R5,WRITEI
4460 026472  000072                    NPRAOX
4461 026474  000200                    NPRBS7                  ;  (THIS SETS BS7 & CLEARS EXTENDED ADDR. BITS)
4462 026476  103003                    BCC     .+10            ;IF NO ERROR, PROCEED
```

TEST 6 NPR XFER ABORT

```
4463 026500                          ERROR                  ;ELSE, REPORT IT
     026500  104460                                                        TRAP    C$ERROR
4464 026502                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     026502  104410                                                        TRAP    C$ESCAPE
     026504  000062                                                        .WORD   L10040-.
4465 026506  004537  004042          JSR     R5,WRITE       ;INITIATE THE NPR-OUT OPERATION
4466 026512  123004                  NPRCTL
4467 026514  002414                  TMP1
4468 026516  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4469 026520                          ERROR                  ;ELSE, REPORT IT
     026520  104460                                                        TRAP    C$ERROR
4470 026522                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     026522  104410                                                        TRAP    C$ESCAPE
     026524  000042                                                        .WORD   L10040-.
4471 026526  004537  003616          JSR     R5,READ        ;READ BACK THE CONTROL-STATUS REGISTER
4472 026532  123004                  NPRCTL
4473 026534  002416                  TMP2
4474 026536  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4475 026540                          ERROR                  ;ELSE, REPORT IT
     026540  104460                                                        TRAP    C$ERROR
4476 026542                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     026542  104410                                                        TRAP    C$ESCAPE
     026544  000022                                                        .WORD   L10040-.
4477 026546  132737  000200  002416  BITB    @NPRABT,TMP2   ;THE ABORT BIT SHOULD BE SET
4478 026554  001004                  BNE     20$            ;IT IS.  EXIT SUBTEST
4479 026556                          GEDF    EM26G,ERR11    ;IT DIDN'T.  REPORT MISSING NPR ABORT
                                                            ;       "DEVICE FATAL" ERROR # 31
     026556  104455                                                        TRAP    C$ERDF
     026560  000037                                                        .WORD   31
     026562  016402                                                        .WORD   EM26G
     026564  007314                                                        .WORD   ERR11
4480 026566                  20$:    ENDSUB                                          L10040:
     026566                                                                 TRAP    C$ESUB
     026566  104403
4481                         ;***  SUBTEST #2 -- NPR OUTPUT TO EXISTENT LOCATION YIELDING NO NPR-ABORT  ****
4482
4483 026570                          BGNSUB                                          T6.2:
     026570                                                                 TRAP    C$BSUB
     026570  104402
4484 026572  004737  003514          JSR     PC,MSTCLR      ;INIT DMV & ENTER M-LOOP
4485 026576  103003                  BCC     .+10           ;IF NO ERROR, PROCEED WITH TESTING
4486 026600                          ERROR                  ;ELSE, REPORT ERROR
     026600  104460                                                        TRAP    C$ERROR
4487 026602                          ESCAPE  SUB            ;   & EXIT TEST
     026602  104410                                                        TRAP    C$ESCAPE
     026604  000164                                                        .WORD   L10041 .
4488 026606  012737  000001  002412  MOV     #1,TMPO        ;DISABLE PRINTOUT OF TIMEOUT-COUNT BY ERR11
4489
4490 026614  012737  002654  002420  MOV     #BUFAREA,TMP3  ;SETUP 11'S ADDRESS
4491 026622  012737  000044  002414  MOV     #NPRDL,TMP1    ;CONTROL REG. VALUE FOR NPR-OUT COMMAND
4492 026630  004537  004042          JSR     R5,WRITE       ;SETUP ADDRESS OUT REGISTERS
4493 026634  000070                  NPRAOL
4494 026636  002420                  TMP3
4495 026640  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4496 026642                          ERROR                  ;ELSE, REPORT IT
     026642  104460                                                        TRAP    C$ERROR
4497 026644                          ESCAPE  SUB            ;       AND EXIT THIS TEST
```

TEST 6 -- NPR XFER ABORT

```
        026644  104410                                                          TRAP    C$ESCAPE
        026646  000122                                                          .WORD   L10041-.
4498    026650  004537  004042          JSR     R5,WRITE
4499    026654  000071                  NPRAOH
4500    026656  002421                  TMP3+1
4501    026660  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
4502    026662                          ERROR                   ;ELSE, REPORT IT
        026662  104460                                                          TRAP    C$ERROR
4503    026664                          ESCAPE  SUB             ;       AND EXIT THIS TEST
        026664  104410                                                          TRAP    C$ESCAPE
        026666  000102                                                          .WORD   L10041-.
4504    026670  004537  004054          JSR     R5,WRITEI
4505    026674  000072                  NPRAOX
4506    026676  000000                  0                       ;  (THIS CLEARS BS7 & EXTENDED ADDR. BITS)
4507    026700  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
4508    026702                          ERROR                   ;ELSE, REPORT IT
        026702  104460                                                          TRAP    C$ERROR
4509    026704                          ESCAPE  SUB             ;       AND EXIT THIS TEST
        026704  104410                                                          TRAP    C$ESCAPE
        026706  000062                                                          .WORD   L10041-.
4510    026710  004537  004042          JSR     R5,WRITE        ;INITIATE THE NPR-OUT OPERATION
4511    026714  123004                  NPRCTL
4512    026716  002414                  TMP1
4513    026720  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
4514    026722                          ERROR                   ;ELSE, REPORT IT
        026722  104460                                                          TRAP    C$ERROR
4515    026724                          ESCAPE  SUB             ;       AND EXIT THIS TEST
        026724  104410                                                          TRAP    C$ESCAPE
        026726  000042                                                          .WORD   L10041-.
4516    026730  004537  003616          JSR     R5,READ         ;READ BACK THE CONTROL-STATUS REGISTER
4517    026734  123004                  NPRCTL
4518    026736  002416                  TMP2
4519    026740  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
4520    026742                          ERROR                   ;ELSE, REPORT IT
        026742  104460                                                          TRAP    C$ERROR
4521    026744                          ESCAPE  SUB             ;       AND EXIT THIS TEST
        026744  104410                                                          TRAP    C$ESCAPE
        026746  000022                                                          .WORD   L10041-.
4522    026750  132737  000200  002416  BITB    #NPRABT,TMP2    ;THE ABORT BIT SHOULD BE SET
4523    026756  001404                  BEQ     20$             ;IT IS.  EXIT SUBTEST
4524    026760                          GEDF    EM26E,ERR11     ;IT DIDN'T.  REPORT MISSING NPR ABORT
                                                                ;       "DEVICE FATAL" ERROR # 32
        026760  104455                                                          TRAP    C$ERDF
        026762  000040                                                          .WORD   32
        026764  016316                                                          .WORD   EM26E
        026766  007314                                                          .WORD   ERR11
4525    026770                  20$:    ENDSUB
        026770                                                          L10041:
        026770  104403                                                          TRAP    C$ESUB
4526    026772                          ENDTST
        026772                                                          L10037:
        026772  104401                                                          TRAP    C$ETST
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
   4544                                 .SBTTL TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

                                        ;+*********************************************************************
                                        ;*
                                        ;*      TEST 7 -- NPR EXTENDED ADDRESS BIT TEST
                                        ;*
                                        ;* THIS TEST WILL ONLY BE RUN IF THERE IS AT LEAST 32K WORDS OF MEMORY ON THE
                                        ;* SYSTEM. IF THERE IS, THE PROGRAM CHOOSES A LOCATION TO USE IN THE ADDRESS
                                        ;* RANGE 200000-377776 (OCTAL). THEN, THE FOLLOWING 2 SUBTESTS ARE PERFORMED :
                                        ;*
                                        ;* FIRST SUBTEST :
                                        ;* AN INPUT NPR IS PERFORMED AND CHECKED USING THE MEMORY LOCATION, WITH
                                        ;* 125252 FOR DATA. THE PROGRAM CHECKS THAT THE ABORT XFER BIT REMAINS
                                        ;* CLEARED.
                                        ;* SECOND SUBTEST :
                                        ;* AN OUTPUT NPR IS PERFORMED AND CHECKED USING THE MEMORY LOCATION, WITH
                                        ;* 125252 FOR DATA. THE PROGRAM CHECKS THAT THE ABORT XFER BIT REMAINS
                                        ;* CLEARED.
                                        ;*
                                        ;--*********************************************************************
                                        ;
                                        ;       BGNTST
     026774                                                                     T7::
   4545 026774  004737  003514          JSR     PC,MSTCLR       ;INIT DMV & ENTER M-LOOP
   4546 027000  103003                  BCC     1$              ;IF NO ERROR, PROCEED WITH TESTING
   4547 027002                          ERROR                   ;ELSE, REPORT ERROR
        027002  104460                                                  TRAP    C$ERROR
   4548 027004                          EXIT    TST             ;   & EXIT TEST
        027004  104432                                                  TRAP    C$EXIT
        027006  001506                                                  .WORD   L10042-.
   4549 027010  013700  002120  1$:     MOV     L$HIMEM,R0      ;GET LAST VALID "PAR" VALUE FROM SUPERVISOR
   4550 027014  042700  001777          BIC     #1777,R0        ;THESE BITS CORESPOND TO BITS 6 --> 15 OF THE
   4551                                                         ;ACTUAL ADDRESS AND AREN'T OUR CONCERN HERE.
   4552 027020  001002                  BNE     2$              ;IF THE RESULT IS ZERO,
   4553 027022                          EXIT    TST             ;       THERE IS NOTHING TO TEST
        027022  104432                                                  TRAP    C$EXIT
        027024  001470                                                  .WORD   L10042-.
   4554                                                         ;ELSE, PROCEED TO SETUP MMU'S PAR AND DMV'S NPR
   4555                                                         ;       REGISTER MAXIMUM VALUES
   4556
   4557 027026  010037  002436  2$:     MOV     R0,TMPA         ;THIS IS FOR THE MMU
   4558
   4559                                 ;*********************************************************************
   4560
   4561 027032                          12$:    BGNSUB                  ;TEST THE NPR-IN USING EXTENDED ADDR. BITS
     027032                                                                     T7.1:
        027032  104402                                                  TRAP    C$BSUB
   4562 027034  012737  177777  002434          MOV     #177777,TMP9    ;INITIALIZE TEMP 9
   4563 027042  012737  002000  002442          MOV     #BIT10,TMPC     ;INITIALIZE PAGE ADDRESS REG. VARIABLE
   4564 027050  112737  000001  002451          MOVB    #BIT0,TMPF+1    ;INITIALIZE NPR EXTENDED ADDRESS REG. VARIABLE
   4565 027056  004737  030516                  JSR     PC,XMINIT       ;INITIALIZE THE MMU
   4566
   4567                                 ;*************** WRITE/READ/VERIFY NPRAIH,NPRAIL ***************
   4568 027062  004537  004054          JSR     R5,WRITEI       ;POINT NPR REGISTERS TO 0
   4569 027066  000074                  NPRAIL
   4570 027070  000000                  0
   4571 027072  004537  004054          JSR     R5,WRITEI
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
4572 027076  000075                        NPRAIH
4573 027100  000000                        0
4574 027102  004537  003616                JSR     R5,READ          ;READ BACK THE ADDRESS & VERIFY IT
4575 027106  000074                        NPRAIL
4576 027110  002434                        TMP9
4577 027112  004537  003616                JSR     R5,READ
4578 027116  000075                        NPRAIH
4579 027120  002435                        TMP9+1
4580 027122  023727  002434  000000        CMP     TMP9,#000000     ;IS IT CORRECT?
4581 027130  001424                        BEQ     2$               ;YES, PROCEED.
4582 027132  013737  002434  002256        MOV     TMP9,BDATA       ;NO. SETUP FOR & REPORT LOADING FAILURE
4583 027140  012737  000000  002254        MOV     #000000,GDATA
4584 027146  012737  000007  002300        MOV     #7,REGNUM        ;IDENTIFY NPRAIH AS THE CULPRIT
4585 027154  105737  002434                TSTB    TMP9             ;IS THAT REALLY TRUE?
4586 027160  001002                        BNE     1$               ;MAYBE.  BUT, NPRAIL IS DEFINITELY AT FAULT
4587 027162  005237  002300                INC     REGNUM           ;        SO IDENTIFY IT AS SUCH
4588 027166                          1$:   GEDF    EM26A,ERR14      ;REPORT THE FAILURE
                                           ;       "DEVICE FATAL" ERROR # 33
     027166  104455                                                        TRAP    C$ERDF
     027170  000041                                                        .WORD   33
     027172  016204                                                        .WORD   EM26A
     027174  011002                                                        .WORD   ERR14
4589 027176                        ESCAPE  SUB                      ;       AND EXIT THIS SUBTEST
     027176  104410                                                        TRAP    C$ESCAPE
     027200  000506                                                        .WORD   L10043-.
4590                               ;*****************************************************************************
4591                               ;****************** MAIN SUBTEST #1 LOOP STARTS HERE *************************
4592                               ;*****************************************************************************
4593
4594                               ;***************** COMPLEMENT OF NPRDRH:NPRDRL => TMP3 ***************
4595 027202  004537  003616          2$:   JSR     R5,READ          ;GET THE CURRENT CONTENTS OF THE NPR DATA REG'S
4596 027206  123000                        NPRDRL
4597 027210  002414                        TMP1
4598 027212  103003                        BCC     .+10             ;IF NO ERROR, PROCEED
4599 027214                                ERROR                    ;ELSE, REPORT IT
     027214  104460                                                        TRAP    C$ERROR
4600 027216                                ESCAPE  SUB              ;       AND EXIT THIS TEST
     027216  104410                                                        TRAP    C$ESCAPE
     027220  000466                                                        .WORD   L10043-.
4601 027222  004537  003616                JSR     R5,READ
4602 027226  123001                        NPRDRH
4603 027230  002415                        TMP1+1
4604 027232  103003                        BCC     .+10             ;IF NO ERROR, PROCEED
4605 027234                                ERROR                    ;ELSE, REPORT IT
     027234  104460                                                        TRAP    C$ERROR
4606 027236                                ESCAPE  SUB              ;       AND EXIT THIS TEST
     027236  104410                                                        TRAP    C$ESCAPE
     027240  000446                                                        .WORD   L10043-.
4607 027242  013737  002414  002420        MOV     TMP1,TMP3        ;USE CURRENT DATA & BUILD BACKGROUND PATTERN
4608 027250  005137  002420                COM     TMP3             ;COMPLEMENT IT TO GENERATE A BACKGROUND PATTERN
4609
4610                               ;********** TMP3 => 1ST LOCATION OF EACH EXTENDED MEMORY BLOCK **********
4611 027254  012737  002000  002446        MOV     #BIT10,TMPE      ;REFILL ALL TEST LOCATIONS STARTING HERE
4612 027262  004537  030636          4$:   JSR     R5,XMWRIT        ;WRITE BACKGROUND PATTERN GENERATED ABOVE
4613 027266  002446                        TMPE                     ; POINTER TO "PAR" FORMAT ADDERSS
4614 027270  002420                        TMP3                     ; POINTER TO DATA (TO BE WRITTEN)
4615 027272  103003                        BCC     .+10             ;IF NO ERROR, PROCEED
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
4616 027274                          ERROR                  ;ELSE, REPORT IT
     027274  104460                                                 TRAP    C$ERROR
4617 027276                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     027276  104410                                                 TRAP    C$ESCAPE
     027300  000406                                                 .WORD   L10043-.
4618 027302  062737  002000  002446  ADD     #BIT10,TMPE    ;INCREMENT THE PAGE ADDR. REG. VALUE
4619 027310  001404                  BEQ     6$             ;DONE IF IT GOES TO ZERO
4620 027312  023737  002446  002436  CMP     TMPE,TMPA      ;IS THE NEW VALUE WITHIN CURRENT MEMORY?
4621 027320  101760                  BLOS    4$             ;YES, THE WRITE IT TOO.
4622                                                        ;NO,  DONE.
4623                          ;*************** WRITE/READ/VERIFY NPRAIX ***************
4624 027322  004537  004042  6$:     JSR     R5,WRITE       ;SETUP NPR EXTENDED ADDR. REG BITS
4625 027326  000076                  NPRAIX
4626 027330  002451                  TMPF+1
4627 027332  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4628 027334                          ERROR                  ;ELSE, REPORT IT
     027334  104460                                                 TRAP    C$ERROR
4629 027336                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     027336  104410                                                 TRAP    C$ESCAPE
     027340  000346                                                 .WORD   L10043-.
4630 027342  004537  003616          JSR     R5,READ        ;READ IT BACK & VERIFY THAT IT'S CORRECT
4631 027346  000076                  NPRAIX
4632 027350  002450                  TMPF
4633 027352  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4634 027354                          ERROR                  ;ELSE, REPORT IT
     027354  104460                                                 TRAP    C$ERROR
4635 027356                          ESCAPE  SUB            ;       AND EXIT THIS TEST
     027356  104410                                                 TRAP    C$ESCAPE
     027360  000326                                                 .WORD   L10043-.
4636 027362  123737  002450  002451  CMPB    TMPF,TMPF+1    ;DID IT LOAD CORRECTLY?
4637 027370  001417                  BEQ     8$             ;YES, PROCEED
4638 027372  113737  002451  002254  MOVB    TMPF+1,GDATA   ;NO, SETUP FOR ERROR HANDLER
4639 027400  113737  002450  002256  MOVB    TMPF,BDATA
4640 027406  012737  000006  002300  MOV     #6,REGNUM      ;       IDENTIFY NPRAIX AS FAILING REG.
4641 027414                          GEDF    EM26A,ERR14    ;REPORT THE FAILURE
                                                            ;       "DEVICE FATAL" ERROR # 34
     027414  104455                                                 TRAP    C$ERDF
     027416  000042                                                 .WORD   34
     027420  016204                                                 .WORD   EM26A
     027422  011002                                                 .WORD   ERR14
4642 027424                          ESCAPE  SUB            ;       AND EXIT THIS SUBTEST
     027424  104410                                                 TRAP    C$ESCAPE
     027426  000260                                                 .WORD   L10043-.
4643
4644                          ;*************** GENERATE/WRITE TEST WORD INTO "TMPC" (LSI-11) ***************
4645 027430  013737  002420  002254  8$:     MOV     TMP3,GDATA     ;GENERATE A TEST DATA PATTERN FROM BACKGROUND
4646 027436  062737  125252  002254          ADD     #125252,GDATA  ;PATTERN BY ADDING THIS TO IT.
4647
4648 027444  004537  030636          JSR     R5,XMWRIT      ;LOAD UP THE TEST PATTERN
4649 027450  002442                  TMPC                   ; POINTER TO "PAR" FORMAT ADDRESS
4650 027452  002254                  GDATA                  ; POINTER TO DATA (TO BE WRITTEN)
4651 027454  103003                  BCC     .+10           ;IF NO ERROR, PROCEED
4652 027456                          ERROR                  ;ELSE, REPORT IT
     027456  104460                                                 TRAP    C$ERROR
4653 027460                          ESCAPE  SUB            ;       AND EXIT THIS SUBTEST
     027460  104410                                                 TRAP    C$ESCAPE
     027462  000224                                                 .WORD   L10043-.
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
4654 027464  112737  000004  002417          MOVB    #NPRLD,TMP2+1   ;SETUP CONTROL VALUE TO DO NPR-IN
4655 027472  004537  004042                  JSR     R5,WRITE        ;PERFORM THE "EXTENDED" DATA-IN NPR
4656 027476  123004                          NPRCTL
4657 027500  002417                          TMP2+1
4658 027502  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4659 027504                                  ERROR                   ;ELSE, REPORT IT
     027504  104460                                                              TRAP    C$ERROR
4660 027506                                  ESCAPE  SUB             ;       AND EXIT THIS SUBTEST
     027506  104410                                                              TRAP    C$ESCAPE
     027510  000176                                                              .WORD   L10043-.
4661
4662                                ;*************** CHECK THE NPR OPERATION (DATA/NPRCTL) ***************
4663 027512  004537  003616                  JSR     R5,READ         ;CHECK THE NPR OPERATION
4664 027516  123004                          NPRCTL
4665 027520  002416                          TMP2
4666 027522  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4667 027524                                  ERROR                   ;ELSE, REPORT IT
     027524  104460                                                              TRAP    C$ERROR
4668 027526                                  ESCAPE  SUB             ;       AND EXIT THIS TEST
     027526  104410                                                              TRAP    C$ESCAPE
     027530  000156                                                              .WORD   L10043-.
4669 027532  004537  003616                  JSR     R5,READ         ;GET THE DATA WE SHOULD HAVE JUST LOADED INTO
4670 027536  123001                          NPRDRH                  ;       THE NPR DATA REGISTERS FROM THE
4671 027540  002257                          BDATA+1                 ;       EXTENDED MEMORY AREA
4672 027542  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4673 027544                                  ERROR                   ;ELSE, REPORT IT
     027544  104460                                                              TRAP    C$ERROR
4674 027546                                  ESCAPE  SUB             ;       AND EXIT THIS TEST
     027546  104410                                                              TRAP    C$ESCAPE
     027550  000136                                                              .WORD   L10043-.
4675 027552  004537  003616                  JSR     R5,READ
4676 027556  123000                          NPRDRL
4677 027560  002256                          BDATA
4678 027562  103003                          BCC     .+10            ;IF NO ERROR, PROCEED
4679 027564                                  ERROR                   ;ELSE, REPORT IT
     027564  104460                                                              TRAP    C$ERROR
4680 027566                                  ESCAPE  SUB             ;       AND EXIT THIS TEST
     027566  104410                                                              TRAP    C$ESCAPE
     027570  000116                                                              .WORD   L10043-.
4681 027572  132737  000300  002416          BITB    #300,TMP2       ;DID IT ABORT OR HANG?
4682 027600  001414                          BEQ     14$             ;NO, GOOD.  PROCEED WITH SUBTEST
4683 027602  100005                          BPL     10$             ;YES, WHICH ONE?
4684 027604                                  GEDF    EM27A,ERR12     ;ABORT, REPORT IT AS SUCH.
                                                                     ;       "DEVICE FATAL" ERROR # 35
     027604  104455                                                              TRAP    C$ERDF
     027606  000043                                                              .WORD   35
     027610  016437                                                              .WORD   EM27A
     027612  007456                                                              .WORD   ERR12
4685 027614  000404                          BR      12$             ;AND EXIT
4686 027616                          10$:    GEDF    EM27B,ERR12     ;HANG, REPORT IT AS SUCH.
                                                                     ;       "DEVICE FATAL" ERROR # 36
     027616  104455                                                              TRAP    C$ERDF
     027620  000044                                                              .WORD   36
     027622  016461                                                              .WORD   EM27B
     027624  007456                                                              .WORD   ERR12
4687 027626                          12$:    ESCAPE  SUB             ;  AND EXIT SUBTEST
     027626  104410                                                              TRAP    C$ESCAPE
```

```
     027630  000056                                                                  .WORD   L10043-.
4688 027632  023737  002256  002254  14$:    CMP     BDATA,GDATA     ;DID WE READ THE TEST DATA USING THE NPR?
4689 027640  001406                          BEQ     20$             ;YES,  WELL THIS ONE WORKED.
4690 027642                                  GEDF    EM27C,ERR12     ;NO!   REPORT THE ERROR.
                                                                     ;       "DEVICE FATAL" ERROR # 37
     027642  104455                                                                  TRAP    C$ERDF
     027644  000045                                                                  .WORD   37
     027646  016502                                                                  .WORD   EM27C
     027650  007456                                                                  .WORD   ERR12
4691 027652                                  ESCAPE  SUB             ;  EXIT FROM SUBTEST AFTER PRINTING ERROR MSG.
     027652  104410                                                                  TRAP    C$ESCAPE
     027654  000032                                                                  .WORD   L10043-.
4692 027656  062737  002000  002442  20$:    ADD     #BIT10,TMPC     ;POINT TO NEXT PAGE ADDRESS REG. VALUE
4693 027664  001410                          BEQ     63$             ;IF 0, WE'RE DONE
4694 027666  023737  002442  002436          CMP     TMPC,TMPA       ;IF GREATER THEN MAXIMUM VALUE,
4695 027674  101004                          BHI     63$             ;       WE'RE DONE TOO.
4696 027676  105237  002451                  INCB    TMPF+1          ;ELSE, INCREMENT NPR'S EXTENDED ADDR. REG.
4697 027702  000137  027202                  JMP     2$              ;AND GO BACK TO DO THIS ADDRESS
4698
4699 027706                          63$:    ENDSUB
     027706                                                                  L10043:
     027706  104403                                                                  TRAP    C$ESUB
4700                                 ;=============================================================================
4701
4702 027710                                  BGNSUB                  ;TEST THE NPR-OUT USING EXTENDED ADDR. BITS
     027710                                                                  T7.2:
     027710  104402                                                                  TRAP    C$BSUB
4703 027712  012737  002000  002442          MOV     #BIT10,TMPC     ;INITIALIZE PAGE ADDRESS REG. VARIABLE
4704 027720  112737  000001  002451          MOVB    #BIT0,TMPF+1    ;INITIALIZE NPR EXTENDED ADDRESS REG. VARIABLE
4705 027726  004737  030516                  JSR     PC,XMINIT       ;INITIALIZE THE MMU
4706
4707                                 ;***************** WRITE/READ/VERIFY NPRAOH,NPRAOL ****************
4708 027732  004537  004054                  JSR     R5,WRITEI       ;POINT NPR REGISTERS TO 0
4709 027736  000070                          NPRAOL
4710 027740  000000                          0
4711 027742  004537  004054                  JSR     R5,WRITEI
4712 027746  000071                          NPRAOH
4713 027750  000000                          0
4714 027752  004537  003616                  JSR     R5,READ         ;READ BACK THE ADDRESS & VERIFY IT
4715 027756  000070                          NPRAOL
4716 027760  002434                          TMP9
4717 027762  004537  003616                  JSR     R5,READ
4718 027766  000071                          NPRAOH
4719 027770  002435                          TMP9+1
4720 027772  023727  002434  000000          CMP     TMP9,#000000    ;IS IT CORRECT?
4721 030000  001427                          BEQ     2$              ;YES, PROCEED.
4722 030002  013737  002434  002256          MOV     TMP9,BDATA      ;NO, SETUP FOR & REPORT LOADING FAILURE
4723 030010  012737  000000  002254          MOV     #000000,GDATA
4724 030016  012737  000004  002300          MOV     #4,REGNUM       ;IDENTIFY NPRAIH AS THE CULPRIT
4725 030024  105737  002434                  TSTB    TMP9            ;IS THAT REALLY TRUE?
4726 030030  001002                          BNE     1$              ;MAYBE.  BUT, NPRAIL IS DEFINATELY AT FAULT
4727 030032  005237  002300                  INC     REGNUM          ;       SO IDENTIFY IT AS SUCH
4728 030036                          1$:     GEDF    EM26A,ERR14     ;REPORT THE FAILURE
                                                                     ;       "DEVICE FATAL" ERROR # 38
     030036  104455                                                                  TRAP    C$ERDF
     030040  000046                                                                  .WORD   38
     030042  016204                                                                  .WORD   EM26A
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
      030044  011002                                                          .WORD   ERR14
 4729 030046                          ESCAPE  SUB             ;       AND EXIT THIS SUBTEST
      030046  104410                                                          TRAP    C$ESCAPE
      030050  000442                                                          .WORD   L10044-.
 4730 030052  012737  123456  002420         MOV     #123456,TMP3    ;USE THIS AS INITIAL BACKGROUND PATTERN
 4731
 4732                              ;*************************************************************************
 4733                              ;***************** MAIN SUBTEST #2 LOOP STARTS HERE ***********************
 4734                              ;*************************************************************************
 4735
 4736                              ;********** TMP3 => 1ST LOCATION OF EACH EXTENDED MEMORY BLOCK **********
 4737 030060  062737  021475  002420  2$:    ADD     #21475,TMP3     ;GENERATE THE PATTERN WE'LL USE THIS TIME
 4738 030066  013737  002420  002414         MOV     TMP3,TMP1       ;PUT HERE FOR ERROR HANDLER
 4739 030074  005137  002414                 COM     TMP1
 4740
 4741 030100  012737  002000  002446         MOV     #BIT10,TMPE     ;REFILL ALL TEST LOCATIONS STARTING HERE
 4742 030106  004537  030636           4$:    JSR     R5,XMWRIT       ;WRITE BACKGROUND PATTERN GENERATED ABOVE
 4743 030112  002446                         TMPE                    ; POINTER TO ADDRESS (IN "PAR" FORMAT)
 4744 030114  002420                         TMP3                    ; POINTER TO DATA (TO BE WRITTEN)
 4745 030116  103003                         BCC     .+10            ;IF NO ERROR, PROCEED
 4746 030120                                 ERROR                   ;ELSE, REPORT IT
      030120  104460                                                          TRAP    C$ERROR
 4747 030122                          ESCAPE  SUB             ;       AND EXIT THIS TEST
      030122  104410                                                          TRAP    C$ESCAPE
      030124  000366                                                          .WORD   L10044-.
 4748 030126  062737  002000  002446         ADD     #BIT10,TMPE     ;INCREMENT THE PAGE ADDR. REG. VALUE
 4749 030134  001404                         BEQ     6$              ;DONE IF IT GOES TO ZERO
 4750 030136  023737  002446  002436         CMP     TMPE,TMPA       ;IS THE NEW VALUE WITHIN CURRENT MEMORY?
 4751 030144  101760                         BLOS    4$              ;YES, THE WRITE IT TOO.
 4752                                                                 ;NO,  DONE.
 4753
 4754                              ;*************** WRITE/READ/VERIFY NPRAIX ***************
 4755 030146  004537  004042           6$:    JSR     R5,WRITE        ;SETUP NPR EXTENDED ADDR. REG BITS
 4756 030152  000072                         NPRAOX
 4757 030154  002451                         TMPF+1
 4758 030156  103003                         BCC     .+10            ;IF NO ERROR, PROCEED
 4759 030160                                 ERROR                   ;ELSE, REPORT IT
      030160  104460                                                          TRAP    C$ERROR
 4760 030162                          ESCAPE  SUB             ;       AND EXIT THIS TEST
      030162  104410                                                          TRAP    C$ESCAPE
      030164  000326                                                          .WORD   L10044-.
 4761 030166  004537  003616                 JSR     R5,READ         ;READ IT BACK & VERIFY THAT IT'S CORRECT
 4762 030172  000072                         NPRAOX
 4763 030174  002450                         TMPF
 4764 030176  103003                         BCC     .+10            ;IF NO ERROR, PROCEED
 4765 030200                                 ERROR                   ;ELSE, REPORT IT
      030200  104460                                                          TRAP    C$ERROR
 4766 030202                          ESCAPE  SUB             ;       AND EXIT THIS TEST
      030202  104410                                                          TRAP    C$ESCAPE
      030204  000306                                                          .WORD   L10044-.
 4767 030206  123737  002450  002451         CMPB    TMPF,TMPF+1     ;DID IT LOAD CORRECTLY?
 4768 030214  001417                         BEQ     8$              ;YES, PROCEED
 4769 030216  113737  002451  002254         MOVB    TMPF+1,GDATA    ;NO, SETUP FOR ERROR HANDLER
 4770 030224  113737  002450  002256         MOVB    TMPF,BDATA
 4771 030232  012737  000003  002300         MOV     #3,REGNUM       ;       IDENTIFY NPRAIX AS FAILING REG.
 4772 030240                                 GEDF    EM26A,ERR14     ;REPORT THE FAILURE
                                                                     ;       "DEVICE FATAL" ERROR # 39
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
        030240  104455                                                          TRAP    C$ERDF
        030242  000047                                                          .WORD   39
        030244  016204                                                          .WORD   EM26A
        030246  011002                                                          .WORD   ERR14
   4773 030250                          ESCAPE  SUB             ;       AND EXIT THIS SUBTEST
        030250  104410                                                          TRAP    C$ESCAPE
        030252  000240                                                          .WORD   L10044-.
   4774
   4775                         ;*************** WRITE(LSI-11) TEST LOCATION BACKGROUND PATTERN ***************
   4776 030254  004537  030636  8$:     JSR     R5,XMWRIT       ;SETUP TEST LOCATION'S BACKGROUND PATTERN
   4777 030260  002442                  TMPC
   4778 030262  002414                  TMP1
   4779 030264  013737  002420  002254          MOV     TMP3,GDATA      ;GENERATE A TEST DATA PATTERN FROM BACKGROUND
   4780 030272  062737  052525  002254          ADD     #52525,GDATA    ;PATTERN BY ADDING THIS TO IT.
   4781
   4782                         ;*************** LOAD(DMV) TEST PATTERN ***************
   4783 030300  004537  004042          JSR     R5,WRITE                ;LOAD UP THE TEST PATTERN
   4784 030304  123001                  NPRDRH
   4785 030306  002255                  GDATA+1
   4786 030310  004537  004042          JSR     R5,WRITE
   4787 030314  123000                  NPRDRL
   4788 030316  002254                  GDATA
   4789 030320  112737  000044  002417          MOVB    #NPRDL,TMP2+1   ;SETUP CONTROL VALUE TO DO NPR-OUT
   4790
   4791                         ;*************** PERFORM/CHECK EXTENDED NPR OPERATION ***************
   4792 030326  004537  004042          JSR     R5,WRITE        ;PERFORM THE "EXTENDED" DATA-OUT NPR
   4793 030332  123004                  NPRCTL
   4794 030334  002417                  TMP2+1
   4795 030336  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
   4796 030340                          ERROR                   ;ELSE, REPORT IT
        030340  104460                                                          TRAP    C$ERROR
   4797 030342                          ESCAPE  SUB             ;       AND EXIT THIS SUBTEST
        030342  104410                                                          TRAP    C$ESCAPE
        030344  000146                                                          .WORD   L10044-.
   4798 030346  004537  003616          JSR     R5,READ         ;CHECK THE NPR OPERATION
   4799 030352  123004                  NPRCTL
   4800 030354  002416                  TMP2
   4801 030356  103003                  BCC     .+10            ;IF NO ERROR, PROCEED
   4802 030360                          ERROR                   ;ELSE, REPORT IT
        030360  104460                                                          TRAP    C$ERROR
   4803 030362                          ESCAPE  SUB             ;       AND EXIT THIS SUBTEST
        030362  104410                                                          TRAP    C$ESCAPE
        030364  000126                                                          .WORD   L10044-.
   4804 030366  132737  000300  002416          BITB    #300,TMP2       ;DID IT ABORT OR HANG?
   4805 030374  001414                  BEQ     14$             ;NO, GOOD.  PROCEED WITH SUBTEST
   4806 030376  100005                  BPL     10$             ;YES, WHICH ONE?
   4807 030400                          GEDF    EM27A,ERR12     ;ABORT, REPORT IT AS SUCH.
                                                                ;       "DEVICE FATAL" ERROR # 40
        030400  104455                                                          TRAP    C$ERDF
        030402  000050                                                          .WORD   40
        030404  016437                                                          .WORD   EM27A
        030406  007456                                                          .WORD   ERR12
   4808 030410  000404                  BR      12$             ;AND EXIT
   4809 030412                  10$:    GEDF    EM27B,ERR12     ;HANG, REPORT IT AS SUCH.
                                                                ;       "DEVICE FATAL" ERROR # 41
        030412  104455                                                          TRAP    C$ERDF
        030414  000051                                                          .WORD   41
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
        030416  016461                                                          .WORD   EM27B
        030420  007456                                                          .WORD   ERR12
   4810 030422                          12$:    ESCAPE  SUB             ;  AND EXIT SUBTEST
        030422  104410                                                          TRAP    C$ESCAPE
        030424  000066                                                          .WORD   L10044-.
   4811 030426  004537  030734          14$:    JSR     R5,XMREAD       ;GET THE DATA WE SHOULD HAVE JUST LOADED INTO
   4812 030432  002442                          TMPC
   4813 030434  002256                          BDATA
   4814 030436  023737  002256  002254          CMP     BDATA,GDATA     ;DID WE READ THE TEST DATA USING THE NPR?
   4815 030444  001406                          BEQ     20$             ;YES.  WELL THIS ONE WORKED.
   4816 030446                                  GEDF    EM27C,ERR12     ;NO!   REPORT THE ERROR.
                                                                        ;       "DEVICE FATAL" ERROR # 42
        030446  104455                                                          TRAP    C$ERDF
        030450  000052                                                          .WORD   42
        030452  016502                                                          .WORD   EM27C
        030454  007456                                                          .WORD   ERR12
   4817 030456                                  ESCAPE  SUB             ;  EXIT FROM SUBTEST AFTER PRINTING ERROR MSG.
        030456  104410                                                          TRAP    C$ESCAPE
        030460  000032                                                          .WORD   L10044-.
   4818 030462  062737  002000  002442  20$:    ADD     #BIT10,TMPC     ;POINT TO NEXT PAGE ADDRESS REG. VALUE
   4819 030470  001410                          BEQ     63$             ;IF 0, WE'RE DONE
   4820 030472  023737  002442  002436          CMP     TMPC,TMPA       ;IF GREATER THEN MAXIMUM VALUE,
   4821 030500  101004                          BHI     63$             ;       WE'RE DONE TOO.
   4822 030502  105237  002451                  INCB    TMPF+1          ;ELSE, INCREMENT NPR'S EXTENDED ADDR. REG.
   4823 030506  000137  030060                  JMP     2$              ;AND GO BACK TO DO THIS ADDRESS
   4824
   4825
   4826 030512                          63$:    ENDSUB
        030512                                                                  L10044:
        030512  104403                                                                  TRAP    C$ESUB
   4827 030514                                  ENDTST
        030514                                                                  L10042:
        030514  104401                                                                  TRAP    C$ETST
   4828
   4829                                 ;*************************************************************************
   4830                                 ; XMINIT -- SUBROUTINE TO INITIALIZE EXTENDED MEMORY (ALIAS: MEMORY MANAGEMENT
   4831                                 ;       UNIT) HARDWARE REGISTERS.
   4832                                 ;
   4833                                 ;*************************************************************************
   4834
   4835 030516  010C46                  XMINIT: MOV     R0,-(SP)        ;SAVE WORKING REGISTERS
   4836 030520  010146                          MOV     R1,-(SP)
   4837 030522  010346                          MOV     R3,-(SP)
   4838 030524  013737  000004  031060          MOV     @#4,XM4HOL      ;SETUP @4 TRAP VALUE (JUST IN CASE)
   4839
   4840 030532  012700  077406                  MOV     #77406,R0       ;"PDR" INITIALIZATION VALUE
   4841                                                                 ;  774 = FULL PAGE ACCESS
   4842                                                                 ;  0 = UPWARD EXPANSION
   4843                                                                 ;  6 = RESIDENT READ/WRITE
   4844 030536  012701  172300                  MOV     #172300,R1      ;ADDRESS OF KPDR0
   4845 030542  012703  000010                  MOV     #8.,R3          ;LOOP VALUE -- # OF PDR'S
   4846 030546  010021                  1$:     MOV     R0,(R1)+        ;SETUP 1 PDR
   4847 030550  077302                          SOB     R3,1$           ;IF ANOTHER PDR, DO IT TOO
   4848                                                                 ;ELSE, FALL THROUGH & INITIALIZE PAR'S
   4849
   4850 030552  005000                          CLR     R0              ;INITIALIZATION VALUE FOR KPAR0
   4851 030554  012701  172340                  MOV     #172340,R1      ;ADDRESS OF KPAR0
```

TEST 7 -- NPR EXTENDED ADDRESS BIT TEST

```
4852 030560  012703  000007              MOV     #7,R3           ;LOOP VALUE -- ONLY FIRST 7 PAR'S DONE BY LOOP
4853 030564  010021                2$:   MOV     R0,(R1)+        ;SETUP 1 PAR
4854 030566  062700  000200              ADD     #200,R0         ;CALCULATE NEXT PAR'S INITIALIZATION VALUE
4855 030572  077304                      SOB     R3,2$           ;IF ANOTHEER PAR, DO IT TOO
4856 030574  012721  177600              MOV     #177600,(R1)+   ;ELSE, SETUP KPAR7 FOR I/O PAGE ACCESSING
4857
4858 030600                              SETVEC  #250,#XMINTH,#7 ;SETUP OUR OWN TRAP CATCHER FOR ABORT HANDLING
     030600  012746  000007                                                  MOV     #7,-(SP)
     030604  012746  031062                                                  MOV     #XMINTH,-(SP)
     030610  012746  000250                                                  MOV     #250,-(SP)
     030614  012746  000003                                                  MOV     #3,-(SP)
     030620  104437                                                          TRAP    C$SVEC
     030622  062706  000010                                                  ADD     #10,SP
4859 030626  012603                      MOV     (SP)+,R3        ;RESTORE CALLER'S REGISTERS
4860 030630  012601                      MOV     (SP)+,R1
4861 030632  012600                      MOV     (SP)+,R0
4862 030634  000207                      RTS     PC              ;RETURN
4863
4864                            ;*************************************************************************
4865                            ; XMWRIT -- SUBROUTINE TO WRITE ONE WORD INTO AN EXTENDED MEMORY LOCATION.
4866                            ;
4867                            ;       CALLING SEQUENCE:
4868                            ;
4869                            ;       JSR     R5,XMWRIT
4870                            ;       <PRINTER TO HIGH ORDER BITS OF ADDRESS IN "PAR" FORMAT>
4871                            ;       <POINTER TO DATA TO BE WRITTEN>
4872                            ;
4873                            ;*************************************************************************
4874
4875 030636  010146          XMWRIT: MOV     R1,-(SP)        ;SAVE REGISTER(S)
4876
4877 030640  012701  172354              MOV     @172354,R1      ;ADDRESS OF KPAR6
4878 030644  011146                      MOV     (R1),-(SP)      ;SAVE CURRENT KPAR6 VALUE
4879 030646  013511                      MOV     @(R5)+,(R1)     ;SETUP "PAR" FOR THIS WRITE
4880 030650  011137  002426              MOV     (R1),TMP6       ;SAVE ADDRESS FOR ERROR MESSAGE
4881 030654  012737  000060  172516      MOV     #BIT4+BIT5,@#172516  ;ENABLE 22 BIT & I/O PAGE ADDRESSING IN SR3
4882 030660  000241                      CLC                     ;CLEAR OUR ERROR FLAG
4883 030664  013737  000004  031060      MOV     @#4,XM4HOL      ;* SETUP TRAP CATCHER @4 (BECAUSE OF MAPPING)
4884 030672  012737  031032  000004      MOV     #XM4INT,@#4     ;*
4885 030700  052737  000001  177572      BIS     #1,@#177572     ;ENABLE MEMORY MANAGEMENT
4886 030706  013537  140000              MOV     @(R5)+,@#140000 ;WRITE ONE WORD IN THE SPECIFIED PAGE
4887 030712  042737  000001  177572      BIC     #1,@#177572     ;TURN OFF MEMORY MANAGEMENT
4888 030720  013737  031060  000004      MOV     XM4HOL,@#4      ;* RESTORE SUPERVISOR TRAP VECTOR @4
4889 030726  012611                      MOV     (SP)+,(R1)      ;RESTORE KPAR6
4890
4891 030730  012601                      MOV     (SP)+,R1        ;RESTORE CALLER'S REGISTER(S)
4892 030732  000205                      RTS     R5              ;RETURN
4893
4894
4895                            ;*************************************************************************
4896                            ; XMREAD -- SUBROUTINE TO READ FROM AN EXTENDED MEMORY LOCATION.
4897                            ;
4898                            ;       CALLING SEQUENCE:
4899                            ;
4900                            ;       JSR     R5,XMREAD
4901                            ;       <PRINTER TO HIGH ORDER BITS OF ADDRESS IN "PAR" FORMAT>
4902                            ;       <POINTER TO DATA RECEIVING LOCATION>
```

```
4903                                    ;
4904                                    ;*****************************************************************************
4905
4906 030734  010146                     XMREAD: MOV     R1,-(SP)        ;SAVE REGISTER(S)
4907
4908 030736  012701  172354                     MOV     #172354,R1      ;ADDRESS OF KPAR6
4909 030742  011146                             MOV     (R1),-(SP)      ;SAVE CURRENT KPAR6 VALUE
4910 030744  013511                             MOV     @(R5)+,(R1)     ;SETUP "PAR" FOR THIS READ
4911 030746  011137  002426                     MOV     (R1),TMP6       ;SAVE ADDRESS FOR ERROR MESSAGE
4912 030752  012737  000060  172516             MOV     #BIT4+BIT5,@#172516  ;ENABLE 22 BIT & I/O PAGE ADDRESSING IN SR3
4913 030760  000241                             CLC                     ;CLEAR OUR ERROR FLAG
4914 030762  013737  000004  031060             MOV     @#4,XM4HOL      ;* SETUP TRAP CATCHER #4 (BECAUSE OF MAPPING)
4915 030770  012737  031032  000004             MOV     #XM4INT,@#4     ;*
4916 030776  052737  000001  177572             BIS     #1,@#177572     ;ENABLE MEMORY MANAGEMENT
4917 031004  013735  140000                     MOV     @#140000,@(R5)+ ;READ ONE WORD IN THE SPECIFIED PAGE
4918 031010  042737  000001  177572             BIC     #1,@#177572     ;TURN OFF MEMORY MANAGEMENT
4919 031016  013737  031060  000004             MOV     XM4HOL,@#4      ;* RESTORE SUPERVISOR TRAP VECTOR #4
4920 031024  012611                             MOV     (SP)+,(R1)      ;RESTORE KPAR6
4921
4922 031026  012601                             MOV     (SP)+,R1        ;RESTORE CALLER'S REGISTER(S)
4923 031030  000205                             RTS     R5              ;RETURN
4924
4925
4926                                    ;*****************************************************************************
4927                                    ; HANDLER FOR #LOC 4 TRAP PROCESSING (FOR TESTS 7 & 8)
4928                                    ;*****************************************************************************
4929 031032  042737  000001  177572     XM4INT: BIC     #1,@#177572     ;TURN OFF MEMORY MANAGEMENT
4930 031040  013737  031060  000004             MOV     XM4HOL,@#4      ;* RESTORE SUPERVISOR TRAP VECTOR #4
4931 031046  000240                             NOP                     ;*
4932 031050  000240                             NOP                     ;*
4933 031052  000240                             NOP                     ;*
4934 031054  000177  146724                     JMP     @#4             ;NOW JUMP THRU IT !
4935 031060  000000                     XM4HOL: 0
4936
4937                                    ;*****************************************************************************
4938                                    ; INTERRUPT HANDLER FOR MEMORY MANAGEMENT ABORT PROCESSING
4939                                    ;*****************************************************************************
4940
4941 031062                                     BGNSRV  XMINTH
     031062                                                                 XMINTH::
4942 031062  010037  002372                     MOV     R0,REG0         ;SAVE GENERAL REGISTERS
4943 031066  010137  002374                     MOV     R1,REG1
4944 031072  010237  002376                     MOV     R2,REG2
4945 031076  010337  002400                     MOV     R3,REG3
4946 031102  010437  002402                     MOV     R4,REG4
4947 031106  010537  002404                     MOV     R5,REG5
4948 031112  016637  000002  002406             MOV     2(SP),REG6      ;SAVE PSW FROM ERROR TRAP
4949 031120  011637  002410                     MOV     (SP),REG7       ;LIKEWISE FOR PC
4950
4951 031124  013737  177572  002412             MOV     @#177572,TMP0   ;SAVE THE MMU'S STATUS/CONTROL REGISTERS
4952 031132  013737  177574  002414             MOV     @#177574,TMP1
4953 031140  013737  177576  002416             MOV     @#177576,TMP2
4954 031146  013737  172516  002420             MOV     @#172516,TMP3
4955
4956 031154  013737  172354  002422             MOV     @#172354,TMP4   ;SAVE KERNEL PAR WE'RE SUPPOSE TO BE USING
4957
4958 031162  013737  172314  002424             MOV     @#172314,TMP5   ;SAVE KERNEL PDR WE'RE SUPPOSE TO BE USING
```

TEST 2 NPR EXTENDED ADDRESS BIT TEST

```
4959
4960 031170  011537  002430                MOV     (R5),TMP2       ;SAVE DATA READ OR WRITTEN
4961
4962 031174                                GTDF    EM27,ERR13      ;QUEUE UP THE MMU ERROR
                                                                   ;       QUEUE "DEVICE FATAL" ERROR # 43
     031174  012737  000001  002202                                          MOV     #T.EDF,ERRTYP
     031202  012737  000053  002204                                          MOV     #43,ERRNBR
     031210  012737  016424  002206                                          MOV     #EM27,ERRMSG
     031216  012737  010240  002210                                          MOV     #ERR13,ERRBLK
4963
4964 031224  052766  000001  000002          BIS     #BIT0,2(SP)     ;SET CARRY BIT (AS ERROR FLAG) IN PSW ON STACK
4965
4966 031232                                ENDSRV
     031232                                                          L10045:
     031232  000002                                                        RTI
4967
```

```
 4986                                .SBTTL TEST 8  - SPECIAL MFG EXTENDED BIT TEST

                                    ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                    ;*
                                    ;*      TEST 8 -- SPECIAL MFG EXTENDED BIT TEST
                                    ;*
                                    ;* THIS TEST WAS DESIGNED SPECIFICALLY TO ALLOW MANUFACTURING TO CHECK THE
                                    ;* NPRAIX/NPRAOX BITS WITHOUT A FULL 4 M. OF MEMORY.
                                    ;*
                                    ;* IT WILL CHECK THE 12 DMV EXTENDED ADDRESS BITS (6:NPRAIX/6:NPRAOX) ON
                                    ;* A Q22 SYSTEM IF MEMORY IS PRESENT AT THE FOLLOWING PHYSICAL ADDRESSES:
                                    ;*
                                    ;*      17600000        17400000        17200000
                                    ;*      16600000        15600000        13600000
                                    ;*       7600000
                                    ;*
                                    ;* FIRST SUBTEST :      TEST "NPRAIX" EXTENDED ADDRESS BITS
                                    ;* SECOND SUBTEST :     TEST "NPRAOX" EXTENDED ADDRESS BITS
                                    ;*
                                    ;*
                                    ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                    ;
                                    ;       BGNTST
    031234                                                                                  T8::
 4987
 4988 031234  032737  000002  002370        BIT     #2,PT.CTL       ;IS THIS A MFG SPECIAL Q22 SYSTEM?
 4989 031242  001002                        BNE     .+6             ;YES: GO START TEST
 4990 031244                                EXIT    TST             ; NO: SKIP THIS TEST
    031244  104432                                                                          TRAP    C$EXIT
    031246  001212                                                                          .WORD   L10046-.
 4991
 4992 031250  004737  003514                JSR     PC,MSTCLR       ;INIT DMV & ENTER M-LOOP
 4993 031254  103003                        BCC     1$              ;IF NO ERROR, PROCEED WITH TESTING
 4994 031256                                ERROR                   ;ELSE, REPORT ERROR
    031256  104460                                                                          TRAP    C$ERROR
 4995 031260                                EXIT    TST             ;   & EXIT TEST
    031260  104432                                                                          TRAP    C$EXIT
    031262  001176                                                                          .WORD   L10046-.
 4996                                   ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 4997                                   ;** SUBTEST #1 : TEST THE NPR-IN EXTENDED ADDRESS BITS
 4998                                   ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 4999 031264                            1$:     BGNSUB
    031264                                                                                  T8.1:
    031264  104402                                                                          TRAP    C$BSUB
 5000 031266  004737  030516                JSR     PC,xMINIT       ;INITIALIZE THE MMU
 5001 031272  005004                        CLR     R4              ;CLEAR INDEX
 5002
 5003 031274  005002                        CLR     R2              ;SETUP EXTENDED MEM BACKGROUND PATTERN
 5004 031276  016262  032462  032500  2$:   MOV     XLOCO(R2),XVALO(R2)     ;(IN XVALO => XVAL6)
 5005 031304  005722                        TST     (R2)+
 5006 031306  020227  000016                CMP     R2,#14.
 5007 031312  001371                        BNE     2$
 5008
 5009                                   ;+++++++++++++++ MAIN LOOP STARTS HERE +++++++++++++++
 5010
 5011                                   ;*      WRITE XVALO, XVAL1, XVAL2, ..., XVAL6 INTO THE SEVEN SPECIFIC
 5012                                   ;*      EXTENDED ADDRESSES SPECIFIED BY XLOCO THRU XLOC6 USING 11/23 MMU
```

```
5013                                    ;*      (XLOC'S SPECIFY THE UPPER TWO BYTES OF THE 3 BYTE EXTENDED ADDR)
5014
5015 031314  005002                             CLR     R2              ;CLEAR LOCAL INDEX
5016 031316  016237  032500  002440  11$:      MOV     XVAL0(R2),TMPB  ;SETUP DATA POINTER
5017 031324  016237  032462  002436            MOV     XLOC0(R2),TMPA  ;SETUP/ADJUST PAR VALUE
5018 031332  006337  002436                     ASL     TMPA
5019 031336  006337  002436                     ASL     TMPA
5020 031342  004537  030636                     JSR     R5,XMWRIT       ;WRITE BACKGROUND PATTERN
5021 031346  002436                             TMPA                    ; POINTER TO PAR VALUE
5022 031350  002440                             TMPB                    ; POINTER TO DATA
5023
5024 031352  005722                             TST     (R2)+           ;BUMP INDEX
5025 031354  020227  000016                     CMP     R2,#14.         ;ALL 'XLOC' EXTENDED ADDRESSES WRITTEN?
5026 031360  001356                             BNE     11$             ; NO:  WRITE ANOTHER
5027
5028                                    ;*************** SETUP DMV'S NPR ADDRESSING REGISTERS ***************
5029                                    ;************** (WRITE/READ/VERIFY NPRAIH,NPRAIL,NPRAIX) *************
5030 031362  116437  032463  031416  3$:      MOVB    XLOC0+1(R4),4$  ;SETUP NPRAIX VALUE.
5031 031370  004537  004054                     JSR     R5,WRITEI       ;POINT NPR REGISTERS TO EXTENDED ADDRESS
5032 031374  000074                             NPRAIL
5033 031376  000000                             0
5034 031400  004537  004054                     JSR     R5,WRITEI
5035 031404  000075                             NPRAIH
5036 031406  000000                             0
5037 031410  004537  004054                     JSR     R5,WRITEI
5038 031414  000076                             NPRAIX
5039 031416  000000                     4$:     00
5040 031420  004537  003616                     JSR     R5,READ         ;READ BACK THE ADDRESS
5041 031424  000074                             NPRAIL
5042 031426  002256                             BDATA
5043 031430  004537  003616                     JSR     R5,READ
5044 031434  000075                             NPRAIH
5045 031436  002257                             BDATA+1
5046 031440  004537  003616                     JSR     R5,READ
5047 031444  000076                             NPRAIX
5048 031446  002434                             TMP9
5049                                                                    ;********** NOW CHECK THEM **********
5050 031450  005737  002256                     TST     BDATA           ;NPRAIL,NPRAIH=0 ?
5051 031454  001413                             BEQ     6$              ; YES: TRY CHECKING NPRAIX
5052 031456  005037  002254                     CLR     GDATA           ; NO:  REPORT ERROR...
5053 031462  012737  000007  002300             MOV     #7,REGNUM       ;
5054 031470  105737  002256                     TSTB    BDATA
5055 031474  001020                             BNE     7$
5056 031476  005237  002300                     INC     REGNUM
5057 031502  000415                             BR      7$
5058 031504  013737  031416  002254  6$:      MOV     4$,GDATA        ;SET UP NPRAIX EXPECTED
5059 031512  013737  002434  002256             MOV     TMP9,BDATA      ;SET UP NPRAIX READ...
5060 031520  023737  002254  002256             CMP     GDATA,BDATA     ;DOES NPRAIX=EXPECTED ?
5061 031526  001411                             BEQ     9$              ; YES: CONTINUE
5062 031530  012737  000006  002300             MOV     #6,REGNUM       ; NO:  REPORT ERROR
5063 031536                             7$:     GEDF    EM26A,ERR14
                                                ;       "DEVICE FATAL" ERROR # 44
     031536  104455                                                            TRAP    C$ERDF
     031540  000054                                                            .WORD   44
     031542  016204                                                            .WORD   EM26A
     031544  011002                                                            .WORD   ERR14
5064 031546                             ESCAPE  SUB
```

```
        031546  104410                                                          TRAP    C$ESCAPE
        031550  000152                                                          .WORD   L10047-.
   5065
   5066                                 ;******* SETUP/START/CHECK THE NPR OPERATION (DATA/NPRCTL) *********
   5067 031552  112737  000004  002417  9$:     MOVB    #NPRLD,TMP2+1   ;SETUP CONTROL VALUE TO DO NPR-IN
   5068
   5069 031560  004537  004042          JSR     R5,WRITE        ;PERFORM THE "EXTENDED" DATA-IN NPR
   5070 031564  123004                  NPRCTL
   5071 031566  002417                  TMP2+1
   5072 031570  004537  003616          JSR     R5,READ         ;CHECK THE NPR OPERATION
   5073 031574  123004                  NPRCTL
   5074 031576  002416                  TMP2
   5075 031600  004537  003616          JSR     R5,READ         ;GET THE DATA WE SHOULD HAVE JUST LOADED INTO
   5076 031604  123001                  NPRDRH                  ;       THE NPR DATA REGISTERS FROM THE
   5077 031606  002257                  BDATA+1                 ;       EXTENDED MEMORY AREA
   5078 031610  004537  003616          JSR     R5,READ
   5079 031614  123000                  NPRDRL
   5080 031616  002256                  BDATA
   5081 031620  132737  000300  002416  BITB    #300,TMP2       ;DID IT ABORT OR HANG?
   5082 031626  001414                  BEQ     14$             ; NO:  GOOD.  PROCEED WITH SUBTEST
   5083 031630  100005                  BPL     10$             ; YES: WHICH ONE?
   5084 031632                          GEDF    EM27A,ERR12     ;ABORT, REPORT IT AS SUCH.
                                                                ;       "DEVICE FATAL" ERROR # 45
        031632  104455                                                          TRAP    C$ERDF
        031634  000055                                                          .WORD   45
        031636  016437                                                          .WORD   EM27A
        031640  007456                                                          .WORD   ERR12
   5085 031642  000404                  BR      12$             ;AND EXIT
   5086 031644                  10$:    GEDF    EM27B,ERR12     ;HANG, REPORT IT AS SUCH.
                                                                ;       "DEVICE FATAL" ERROR # 46
        031644  104455                                                          TRAP    C$ERDF
        031646  000056                                                          .WORD   46
        031650  016461                                                          .WORD   EM27B
        031652  007456                                                          .WORD   ERR12
   5087 031654                  12$:    ESCAPE  SUB             ;  AND EXIT SUBTEST
        031654  104410                                                          TRAP    C$ESCAPE
        031656  000044                                                          .WORD   L10047-.
   5088
   5089                                 ;************** NOW CHECK DATA READ AGAINST EXPECTED VALUE **************
   5090 031660  016437  032462  002254  14$:     MOV     XLOCO(R4),GDATA ;SET UP EXPECTED READ VALUE
   5091
   5092 031666  023737  002256  002254          CMP     BDATA,GDATA     ;DID WE READ THE TEST DATA USING THE NPR?
   5093 031674  001406                  BEQ     15$             ;
   5094 031676                          GEDF    EM27C,ERR12     ; NO:  REPORT THE ERROR.
                                                                ;       "DEVICE FATAL" ERROR # 47
        031676  104455                                                          TRAP    C$ERDF
        031700  000057                                                          .WORD   47
        031702  016502                                                          .WORD   EM27C
        031704  007456                                                          .WORD   ERR12
   5095 031706                          ESCAPE  SUB             ;       AND EXIT FROM SUBTEST
        031706  104410                                                          TRAP    C$ESCAPE
        031710  000012                                                          .WORD   L10047-.
   5096
   5097 031712  005724                  15$:    TST     (R4)+           ; YES: BUMP INDEX
   5098 031714  020427  000016          CMP     R4,#14.         ;ARE WE DONE W/ALL EXTENDED LOCATIONS
   5099 031720  001220                  BNE     3$              ; NO; DO NEXT EXTENDED LOCATION
   5100
```

TEST 8 -- SPECIAL MFG EXTENDED BIT TEST

```
5101 031722                          63$:    ENDSUB                  ;YES: END SUBROUTINE
     031722                                                                          L10047:
     031722  104403                                                                          TRAP    C$ESUB
5102
5103
5104
5105                                 ;*************************************************************************
5106                                 ;**  SUBTEST #2 : TEST THE NPR-OUT EXTENDED ADDRESS BITS
5107                                 ;*************************************************************************
5108 031724                                  BGNSUB                  ;TEST THE NPR-OUT USING EXTENDED ADDR. BITS
     031724                                                                          T8.2:
     031724  104402                                                                          TRAP    C$BSUB
5109 031726  004737  030516                    JSR     PC,XMINIT       ;INITIALIZE THE MMU
5110 031732  005004                           CLR     R4              ;CLEAR INDEX
5111
5112                                 ;************** MAIN LOOP STARTS HERE **************
5113 031734  005002                 T8LP:     CLR     R2              ;SETUP EXTENDED MEM BACKGROUND PATTERN
5114 031736  012762  125252  032500  3$:       MOV     #125252,XVALO(R2)       ;(125252 IN XVALO => XVAL6)
5115 031744  005722                           TST     (R2)+                   ; THIS IS DONE FOR ERROR REPORTING
5116 031746  020227  000016                    CMP     R2,#14.                 ; PURPOSES.
5117 031752  001371                           BNE     3$
5118
5119                                 ;*      WRITE (USING MMU) XVAL0, XVAL1, XVAL2, ... XVAL6 (IE: 125252) INTO
5120                                 ;*      THE SEVEN SPECIFIC EXTENDED ADDRESSES SPECIFIED BY XLOC0 THRU XLOC6
5121                                 ;*      (XLOC'S SPECIFY THE UPPER TWO BYTES OF THE 3 BYTE EXTENDED ADDR)
5122
5123 031754  005002                           CLR     R2              ;CLEAR LOCAL INDEX
5124 031756  016237  032500  002440  11$:      MOV     XVALO(R2),TMPB  ;SETUP DATA POINTER
5125 031764  016237  032462  002436           MOV     XLOC0(R2),TMPA  ;SETUP/ADJUST PAR VALUE
5126 031772  006337  002436                    ASL     TMPA
5127 031776  006337  002436                    ASL     TMPA
5128 032002  004537  030636                    JSR     R5,XMWRIT       ;WRITE BACKGROUND PATTERN INTO EXTENDED MEMORY
5129 032006  002436                           TMPA                    ; POINTER TO PAR VALUE
5130 032010  002440                           TMPB                    ; POINTER TO DATA (@DATA + 125252)
5131
5132 032012  005722                           TST     (R2)+           ;BUMP INDEX
5133 032014  020227  000016                    CMP     R2,#14.         ;ALL 'XLOC' EXTENDED ADDRESSES WRITTEN?
5134 032020  001356                           BNE     11$             ; NO:  WRITE ANOTHER
5135
5136                                 ;* WE NOW CHANGE ONE LOCATION IN THE "XVAL" BACKGROUND TABLE.
5137                                 ;* AFTER OUR DMV NPR OUT, THIS TABLE WILL REPRESENT THE
5138                                 ;* EXPECTED VALUES OF OUR EXTENDED MEMORY.
5139 032022  012764  052525  032500           MOV     #052525,XVALO(R4)       ;SETUP EXPECTED PATTERN AFTER NPR-OUT
5140                                                                          ;XVALO => XVAL6 NOW = EXPECTED PATTERN
5141
5142                                 ;*************** SETUP DMV NPR ADDRESSING REGISTERS ***************
5143                                 ;************ (WRITE/READ/VERIFY NPRAOH,NPRAOL,NPRAOX) *************
5144 032030  116437  032463  032064           MOVB    XLOC0+1(R4),5$  ;INIT NPRAOX VALUE
5145 032036  004537  004054                    JSR     R5,WRITEI       ;POINT NPR REGISTERS TO EXTENDED ADDRESS
5146 032042  000070                           NPRAOL
5147 032044  000000                           0
5148 032046  004537  004054                    JSR     R5,WRITEI
5149 032052  000071                           NPRAOH
5150 032054  000000                           0
5151 032056  004537  004054                    JSR     R5,WRITEI
5152 032062  000072                           NPRAOX
5153 032064  000000                 5$:       00
```

TEST 8 -- SPECIAL MFG EXTENDED BIT TEST

```
5154 032066  004537  003616                    JSR     R5,READ          ;READ BACK THE ADDRESS
5155 032072  000070                            NPRAOL
5156 032074  002256                            BDATA
5157 032076  004537  003616                    JSR     R5,READ
5158 032102  000071                            NPRAOH
5159 032104  002257                            BDATA+1
5160 032106  004537  003616                    JSR     R5,READ
5161 032112  000072                            NPRAOX
5162 032114  002434                            TMP9
5163                                                                    ;********** NOW CHECK THEM **********
5164 032116  005737  002256                    TST     BDATA            ;NPRAOL,NPRAOH=0 ?
5165 032122  001413                            BEQ     6$               ; YES: TRY CHECKING NPRAOX
5166 032124  005037  002254                    CLR     GDATA            ; NO:  REPORT ERROR...
5167 032130  012737  000004  002300            MOV     #4,REGNUM        ;
5168 032136  105737  002256                    TSTB    BDATA
5169 032142  001020                            BNE     7$
5170 032144  005237  002300                    INC     REGNUM
5171 032150  000415                            BR      7$
5172 032152  013737  032064  002254  6$:       MOV     5$,GDATA         ;SET UP NPRAOX EXPECTED
5173 032160  013737  002434  002256            MOV     TMP9,BDATA       ;SET UP NPRAOX READ...
5174 032166  023737  002254  002256            CMP     GDATA,BDATA      ;DOES NPRAOX=EXPECTED ?
5175 032174  001411                            BEQ     9$               ; YES: CONTINUE
5176 032176  012737  000003  002300            MOV     #3,REGNUM        ; NO:  REPORT ERROR
5177 032204                            7$:     GEDF    EM26A,ERR14
     032204  104455                                                    ;       "DEVICE FATAL" ERROR # 48
     032206  000060                                                            TRAP    C$ERDF
     032210  016204                                                            .WORD   48
     032212  011002                                                            .WORD   EM26A
5178 032214                                                                    .WORD   ERR14
     032214  104410                    ESCAPE  SUB
     032216  000240                                                            TRAP    C$ESCAPE
5179                                                                           .WORD   L10050-.
5180                                   ;******** SETUP/START/CHECK THE NPR OPERATION (DATA/NPRCTL) *********
5181 032220  012737  052525  002254  9$:       MOV     #052525,GDATA    ;DATA DMV WILL NPR TO TOP LOC
5182
5183                                   ;************** LOAD (DMV) TEST PATTERN *************
5184 032226  004537  004042                    JSR     R5,WRITE                 ;LOAD UP THE TEST PATTERN TO BE
5185 032232  123001                            NPRDRH                           ;WRITTEN INTO EXTENDED MEMORY BY
5186 032234  002255                            GDATA+1                          ;THE DMV
5187 032236  004537  004042                    JSR     R5,WRITE
5188 032242  123000                            NPRDRL
5189 032244  002254                            GDATA
5190
5191                                   ;***************** PERFORM/CHECK NPR OPERATION (BUT NOT DATA) ****************
5192 032246  112737  000044  002417            MOVB    #NPRDL,TMP2+1    ;SETUP CONTROL VALUE TO DO NPR-OUT
5193
5194 032254  004537  004042                    JSR     R5,WRITE         ;PERFORM THE "EXTENDED" DATA-OUT NPR
5195 032260  123004                            NPRCTL
5196 032262  002417                            TMP2+1
5197 032264  004537  003616                    JSR     R5,READ          ;CHECK THE NPR OPERATION
5198 032270  123004                            NPRCTL
5199 032272  002416                            TMP2
5200 032274  132737  000300  002416            BITB    #300,TMP2        ;DID IT ABORT OR HANG?
5201 032302  001414                            BEQ     14$              ;NO, GOOD.  PROCEED WITH SUBTEST
5202 032304  100000                            BPL     10$              ;YES, WHICH ONE?
5203 032306                                    GEDF    EM27A,ERR12      ;ABORT, REPORT IT AS SUCH.
```

```
                                                        ;       "DEVICE FATAL" ERROR # 49
     032306  104455                                                                  TRAP    C$ERDF
     032310  000061                                                                  .WORD   49
     032312  016437                                                                  .WORD   EM27A
     032314  007456                                                                  .WORD   ERR12
5204 032316  000404                 BR      12$             ;AND EXIT
5205 032320                 10$:    GEDF    EM27B,ERR12     ;HANG, REPORT IT AS SUCH.
                                                        ;       "DEVICE FATAL" ERROR # 50
     032320  104455                                                                  TRAP    C$ERDF
     032322  000062                                                                  .WORD   50
     032324  016461                                                                  .WORD   EM27B
     032326  007456                                                                  .WORD   ERR12
5206 032330                 12$:    ESCAPE  SUB             ;  AND EXIT SUBTEST
     032330  104410                                                                  TRAP    C$ESCAPE
     032332  000124                                                                  .WORD   L10050-.
5207
5208                        ;********** READ EXTENDED MEM INTO LOCAL RAM (RXVALO-6) ***************
5209 032334  005002                 14$:    CLR     R2              ;CLEAR LOCAL INDEX
5210 032336  016237  032462  002436 15$:    MOV     XLOCO(R2),TMPA  ;SETUP/ADJUST PAR VALUE
5211 032344  006337  002436                 ASL     TMPA
5212 032350  006337  002436                 ASL     TMPA
5213 032354  004537  030734                 JSR     R5,XMREAD       ;READ EXTENDED MEM BACKGROUND PATTERN
5214 032360  002436                         TMPA                    ; POINTER TO PAR VALUE
5215 032362  002440                         TMPB                    ; POINTER TO DATA STORAGE
5216 032364  013762  002440  032516         MOV     TMPB,RXVALO(R2) ;# SAVE ACTUAL EXTENDED DATA
5217 032372  062702  000002                 ADD     #2,R2           ;BUMP INDEX
5218 032376  020227  000016                 CMP     R2,#14.         ;ALL 'XLOC' EXTENDED ADDRESSES READ?
5219 032402  001355                         BNE     15$             ; NO:  READ ANOTHER
5220
5221                        ;********** NOW CHECK EXPECTED VS. ACTUAL EXT. MEM VALUES **********
5222 032404  005002                 CLR     R2
5223 032406  026262  032516  032500 16$:    CMP     RXVALO(R2),XVALO(R2)
5224 032414  001406                         BEQ     17$
5225 032416                         GEDF    EM60N,ERR60
                                                        ;       "DEVICE FATAL" ERROR # 51
     032416  104455                                                                  TRAP    C$ERDF
     032420  000063                                                                  .WORD   51
     032422  021711                                                                  .WORD   EM60N
     032424  011720                                                                  .WORD   ERR60
5226 032426                         ESCAPE  SUB
     032426  104410                                                                  TRAP    C$ESCAPE
     032430  000026                                                                  .WORD   L10050-.
5227 032432  005722                 17$:    TST     (R2)+           ;BUMP LOCAL INDEX
5228 032434  020227  000016                 CMP     R2,#14.         ;ALL VALUES CHECKED ?
5229 032440  001362                         BNE     16$
5230
5231 032442  005724                 20$:    TST     (R4)+           ; YES; BUMP INDEX
5232 032444  020427  000016                 CMP     R4,#14.         ;ARE WE DONE W/ALL EXTENDED LOCATIONS
5233 032450  001402                         BEQ     63$             ; YES; END
5234 032452  000137  031734                 JMP     T8LP            ; NO:  GO DO SOME MORE
5235 032456                 63$:    ENDSUB
     032456                                                                  L10050:
     032456  104403                                                                  TRAP    C$ESUB
5236 032460                 MFEND:  ENDTST
     032460                                                                  L10046:
     032460  104401                                                                  TRAP    C$ETST
5237
```

TEST 8 -- SPECIAL MFG EXTENDED BIT TEST

```
5238
5239 032462  037400          XLOC0:  37400      ;ADDRESS 17600000 POINTER
5240 032464  037000          XLOC1:  37000      ;ADDRESS 17400000 POINTER
5241 032466  036400          XLOC2:  36400      ;ADDRESS 17200000 POINTER
5242 032470  035400          XLOC3:  35400      ;ADDRESS 16600000 POINTER
5243 032472  033400          XLOC4:  33400      ;ADDRESS 15600000 POINTER
5244 032474  027400          XLOC5:  27400      ;ADDRESS 13600000 POINTER
5245 032476  017400          XLOC6:  17400      ;ADDRESS 07600000 POINTER
5246
5247 032500  000000          XVAL0:  0
5248 032502  000000          XVAL1:  0
5249 032504  000000          XVAL2:  0
5250 032506  000000          XVAL3:  0
5251 032510  000000          XVAL4:  0
5252 032512  000000          XVAL5:  0
5253 032514  000000          XVAL6:  0
5254
5255 032516  000000          RXVAL0: 0
5256 032520  000000          RXVAL1: 0
5257 032522  000000          RXVAL2: 0
5258 032524  000000          RXVAL3: 0
5259 032526  000000          RXVAL4: 0
5260 032530  000000          RXVAL5: 0
5261 032532  000000          RXVAL6: 0
```

```
5282                            .SBTTL TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION

                                ;+*********************************************************************
                                ;*
                                ;*      TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION
                                ;*
                                ;* THIS  TEST  CONTAINS SUBTESTS IN WHICH A SEQUENCE OF STEPS IS
                                ;* PERFORMED. IN GENERAL, EACH SUBTEST PERFORMS THE FOLLOWING:
                                ;*
                                ;*      1. INTERRUPTS ARE DISABLED FOR BOTH "A" & "B"
                                ;*
                                ;*      2. THE INTERRUPT REQUEST REGISTER IS WRITTEN INTO
                                ;*
                                ;*      3. A TEST IS MADE TO BE SURE THAT NEITHER INTERRUPT OCCURS
                                ;*
                                ;*      4. BOTH INTERRUPTS ARE ENABLES
                                ;*
                                ;*      5. A TEST IS MADE TO BE SURE THAT IF AN INTERRUPT IS EXPECTED, IT IS
                                ;*         RECEIVED AND IF IT ISN'T EXPECTED IT DOESN'T HAPPEN.
                                ;*
                                ;* ALL TESTING IS DONE HERE WITH THE PROCESSOR'S PRIORITY SET AT 0.
                                ;*
                                ;--*********************************************************************
                                ;
                                ;       BGNTST
     032534                                                                     T9::
5283 032534  004737  003514            JSR     PC,MSTCLR       ;ISSUE MASTER CLEAR & ENTER MAINT. LOOP
5284 032540  103003                    BCC     1$              ;IF NO ERROR, CONTINUE
5285 032542                            ERROR                   ;ELSE, REPORT IT AND
     032542  104430                                                     TRAP    C$ERROR
5286 032544                            EXIT    TST             ;EXIT THIS TEST
     032544  104432                                                     TRAP    C$EXIT
     032546  001272                                                     .WORD   L10051-.
5287
5288                            ;-------------------------------------------------------------------------
5289                            ;               TEST FOR NO INTERRUPT WHEN ENABLED
5290                            ;
5291                            ;       1. DISABLE BOTH INTERRUPTS
5292                            ;
5293                            ;       2. ASSERT BOTH REQUEST BITS TO 1
5294                            ;
5295                            ;       3. CHECK FOR NO "A" INTERRUPT
5296                            ;
5297                            ;       4. CHECK FOR NO "B" INTERRUPT
5298                            ;
5299                            ;       5. ENABLE BOTH INTERRUPTS
5300                            ;
5301                            ;       6. CHECK FOR NO "A" INTERRUPT
5302                            ;
5303                            ;       7. CHECK FOR NO "B" INTERRUPT
5304
5305 032550                     1$:    BGNSUB
     032550                                                                 T9.1:
     032550  104402                                                             TRAP    C$BSUB
5306 032552  012737  177777  002274        MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
5307 032560  005037  002272              CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5308 032564  112777  000000  147526      MOVB    #0,@BSEL0       ;DISABLE BOTH INTERRUPTS
```

TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION

```
5309 032572  004537  004054              JSR     R5,WRITEI       ;LOAD THE INTERRUPT CONTROL REGISTER WITH
5310 032576  123005                      IRQREG                  ;  BOTH BITS SET.  THIS SHOULD NOT CAUSE
5311 032600  000006                      IRQA!IRQB               ;  AN INTERRUPT AT EITHER LEVEL
5312 032602  103003                      BCC     30$             ;IF AN ERROR OCCURED,
5313 032604                              ERROR                   ;REPORT IT &
     032604  104460                                                      TRAP    C$ERROR
5314 032606                              ESCAPE  TST             ;  QUIT
     032606  104410                                                      TRAP    C$ESCAPE
     032610  001230                                                      .WORD   L10051-.
5315
5316 032612  105737  002272      30$:    TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5317 032616  001407                      BEQ     5$              ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
5318 032620  012737  000001  002254      MOV     #1,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5319 032626                              GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 52
     032626  104455                                                      TRAP    C$ERDF
     032630  000064                                                      .WORD   52
     032632  016526                                                      .WORD   EM34
     032634  006212                                                      .WORD   ERR1
5320
5321 032636  105737  002273      5$:     TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5322 032642  001407                      BEQ     6$              ;NO,  GOOD.  NOW TRY LETTING ONE THROUGH
5323 032644  012737  000002  002254      MOV     #2,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5324 032652                              GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 53
     032652  104455                                                      TRAP    C$ERDF
     032654  000065                                                      .WORD   53
     032656  016540                                                      .WORD   EM34B
     032660  006212                                                      .WORD   ERR1
5325
5326 032662  005037  002272      6$:     CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5327 032666  112777  000021  147424      MOVB    #IENBA!IENBB,@BSELO     ;ENABLE BOTH INTERRUPTS
5328 032674  012703  001000              MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
5329 032700  077301                      SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
5330 032702  105737  002272              TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5331 032706  001407                      BEQ     7$              ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
5332 032710  012737  000003  002254      MOV     #3,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5333 032716                              GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 54
     032716  104455                                                      TRAP    C$ERDF
     032720  000066                                                      .WORD   54
     032722  016526                                                      .WORD   EM34
     032724  006212                                                      .WORD   ERR1
5334
5335 032726  105737  002273      7$:     TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5336 032732  001407                      BEQ     8$              ;NO,  GOOD.  NOW TRY LETTING ONE THROUGH
5337 032734  012737  000004  002254      MOV     #4,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5338 032742                              GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 55
     032742  104455                                                      TRAP    C$ERDF
     032744  000067                                                      .WORD   55
     032746  016540                                                      .WORD   EM34B
     032750  006212                                                      .WORD   ERR1
5339
5340 032752              8$:     ENDSUB
     032752                                                  L10052:
     032752  104403                                                      TRAP    C$ESUB
```

```
5341
5342                            ;-------------------------------------------------------------------------
5343                            ;               TEST FOR "A" INTERRUPT WHEN ENABLED
5344                            ;
5345                            ;       1. DISABLE BOTH INTERRUPTS
5346                            ;
5347                            ;       2. ASSERT "B" REQUEST BIT TO 1:
5348                            ;               DISABLING "B" & FORCING "A"
5349                            ;
5350                            ;       3. CHECK FOR NO "A" INTERRUPT
5351                            ;
5352                            ;       4. CHECK FOR NO "B" INTERRUPT
5353                            ;
5354                            ;       5. ENABLE BOTH INTERRUPTS
5355                            ;
5356                            ;       6. CHECK FOR "A" INTERRUPT
5357                            ;
5358                            ;       7. CHECK FOR NO "B" INTERRUPT
5359
5360 032754                             BGNSUB
      032754                                                                    T9.2:
      032754  104402                                                                   TRAP    C$BSUB
5361 032756  012737  177777  002274      MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
5362 032764  005037  002272              CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5363 032770  112777  000000  147322      MOVB    #0,@BSELO       ;DISABLE INTERRUPTS AGAIN
5364 032776  004537  004054              JSR     R5,WRITEI       ;CAUSE AN INTERRUPT PENDING ON "A"
5365 033002  123005                      IRQREG                  ;   BUT NOT ON "B"
5366 033004  000002                      IRQB
5367 033006  103003                      BCC     31$             ;IF AN ERROR OCCURED,
5368 033010                              ERROR                   ;REPORT IT &
      033010  104460                                                            TRAP    C$ERROR
5369 033012                              ESCAPE  TST             ;  QUIT
      033012  104410                                                            TRAP    C$ESCAPE
      033014  001024                                                            .WORD   L10051-.
5370
5371 033016  105737  002272      31$:    TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5372 033022  001407                      BEQ     10$             ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
5373 033024  012737  000005  002254      MOV     #5,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5374 033032                              GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 56
      033032  104455                                                            TRAP    C$ERDF
      033034  000070                                                            .WORD   56
      033036  016526                                                            .WORD   EM34
      033040  006212                                                            .WORD   ERR1
5375
5376 033042  105737  002273      10$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5377 033046  001407                      BEQ     11$             ;NO,  GOOD.  NOW TRY LETTING ONE THROUGH
5378 033050  012737  000006  002254      MOV     #6,GDATA        ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5379 033056                              GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                 ;       "DEVICE FATAL" ERROR # 57
      033056  104455                                                            TRAP    C$ERDF
      033060  000071                                                            .WORD   57
      033062  016540                                                            .WORD   EM34B
      033064  006212                                                            .WORD   ERR1
5380
5381 033066  005037  002272      11$:    CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5382 033072  112777  000021  147220      MOVB    #IENBA!IENBB,@BSELO     ;ENABLE BOTH INTERRUPTS
```

TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION

```
5383 033100  012703  001000                MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
5384 033104  077301                        SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
5385 033106  105737  002272                TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5386 033112  001007                        BNE     12$             ;YES, GOOD.  GO TEST THE "B" INTERRUPT
5387 033114  012737  000007  002254        MOV     #7,GDATA        ;NO.  TELL ERROR HANDLER WHAT WE HAD DONE
5388 033122                                GEDF    EM35,ERR1       ;REPORT MISSING INTERRUPT ON "ENABLE"
                                                                   ;       "DEVICE FATAL" ERROR # 58
     033122  104455                                                                TRAP    C$ERDF
     033124  000072                                                                .WORD   58
     033126  016552                                                                .WORD   EM35
     033130  006212                                                                .WORD   ERR1
5389
5390 033132  105737  002273          12$:   TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5391 033136  001407                        BEQ     13$             ;NO,  GOOD.  NOW TRY HITTING THE "B" INTERRUPT
5392 033140  012737  000010  002254        MOV     #10,GDATA       ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5393 033146                                GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                   ;       "DEVICE FATAL" ERROR # 59
     033146  104455                                                                TRAP    C$ERDF
     033150  000073                                                                .WORD   59
     033152  016540                                                                .WORD   EM34B
     033154  006212                                                                .WORD   ERR1
5394
5395 033156                         13$:   ENDSUB
     033156                                                                L10053:
     033156  104403                                                                TRAP    C$ESUB
5396
5397
5398                                ;-----------------------------------------------------------------------
5399                                ;               TEST FOR "B" INTERRUPT WHEN ENABLED
5400                                ;
5401                                ;       1. DISABLE BOTH INTERRUPTS
5402                                ;
5403                                ;       2. ASSERT "A" REQUEST BIT TO 1:
5404                                ;            DISABLING "A" & FORCING "B"
5405                                ;
5406                                ;       3. CHECK FOR NO "A" INTERRUPT
5407                                ;
5408                                ;       4. CHECK FOR NO "B" INTERRUPT
5409                                ;
5410                                ;       5. ENABLE BOTH INTERRUPTS
5411                                ;
5412                                ;       6. CHECK FOR NO "A" INTERRUPT
5413                                ;
5414                                ;       7. CHECK FOR "B" INTERRUPT
5415 033160                                BGNSUB
     033160                                                                T9.3:
     033160  104402                                                                TRAP    C$BSUB
5416 033162  012737  177777  002274        MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
5417 033170  005037  002272                CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5418 033174  112777  000000  147116        MOVB    #0,@BSELO       ;DISABLE INTERRUPTS AGAIN
5419 033202  004537  004054                JSR     R5,WRITEI       ;CAUSE AN INTERRUPT PENDING ON "B"
5420 033206  123005                        IRQREG                  ;   BUT NOT ON "A"
5421 033210  000004                        IRQA
5422 033212  103003                        BCC     .+6             ;IF AN ERROR OCCURED,
5423 033214                                ERROR                   ;REPORT IT &
     033214  104460                                                                TRAP    C$ERROR
5424 033216                                ESCAPE  TST             ;  EXIT
```

TEST 9 Q-BUS INTERRUPT "A" & "B" SELECTION

```
                033216  104410                                                          TRAP    C$ESCAPE
                033220  000620                                                          .WORD   L10051-.
          5425
          5426  033222  105737  002272          5$:     TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
          5427  033226  001407                          BEQ     14$             ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
          5428  033230  012737  000011  002254          MOV     #9.,GDATA       ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
          5429  033236                                  GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                                  ;       "DEVICE FATAL" ERROR # 60
                033236  104455                                                          TRAP    C$ERDF
                033240  000074                                                          .WORD   60
                033242  016526                                                          .WORD   EM34
                033244  006212                                                          .WORD   ERR1
          5430
          5431  033246  105737  002273          14$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
          5432  033252  001407                          BEQ     15$             ;NO,  GOOD.  NOW TRY LETTING ONE THROUGH
          5433  033254  012737  000012  002254          MOV     #10.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
          5434  033262                                  GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                                  ;       "DEVICE FATAL" ERROR # 61
                033262  104455                                                          TRAP    C$ERDF
                033264  000075                                                          .WORD   61
                033266  016540                                                          .WORD   EM34B
                033270  006212                                                          .WORD   ERR1
          5435
          5436  033272  005037  002272          15$:    CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
          5437  033276  112777  000021  147014          MOVB    #IENBA!IENBB,@#SELO     ;ENABLE BOTH INTERRUPTS
          5438  033304  012703  001000                  MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
          5439  033310  077301                          SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
          5440  033312  105737  002272                  TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
          5441  033316  001407                          BEQ     16$             ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
          5442  033320  012737  000013  002254          MOV     #11.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
          5443  033326                                  GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                                  ;       "DEVICE FATAL" ERROR # 62
                033326  104455                                                          TRAP    C$ERDF
                033330  000076                                                          .WORD   62
                033332  016526                                                          .WORD   EM34
                033334  006212                                                          .WORD   ERR1
          5444
          5445  033336  105737  002273          16$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
          5446  033342  001007                          BNE     17$             ;YES, GOOD.  NOW TRY HITTING THE "B" INTERRUPT
          5447  033344  012737  000014  002254          MOV     #12.,GDATA      ;NO,  TELL ERROR HANDLER WHAT WE HAD DONE
          5448  033352                                  GEDF    EM35B,ERR1      ;REPORT MISSING INTERRUPT ON "ENABLE"
                                                                                  ;       "DEVICE FATAL" ERROR # 63
                033352  104455                                                          TRAP    C$ERDF
                033354  000077                                                          .WORD   63
                033356  016573                                                          .WORD   EM35B
                033360  006212                                                          .WORD   ERR1
          5449
          5450  033362                          17$:    ENDSUB
                033362                                                          L10054:
                033362  104403                                                          TRAP    C$ESUB
          5451
          5452                                  ;
          5453                                  ;   TEST FOR "A" INTERRUPT BUT NO "B" WHEN BOTH ENABLED & FORCED
          5454                                  ;
          5455                                  ;       1. DISABLE BOTH INTERRUPTS
          5456                                  ;
          5457                                  ;       2. ASSERT BOTH "A" & "B" REQUEST BITS TO 0;
```

```
5458                                    ;           FORCING BOTH "A" & "B" (BUT ONLY GETTING "A")
5459                                    ;
5460                                    ;       3. CHECK FOR NO "A" INTERRUPT
5461                                    ;
5462                                    ;       4. CHECK FOR NO "B" INTERRUPT
5463                                    ;
5464                                    ;       5. ENABLE BOTH INTERRUPTS
5465                                    ;
5466                                    ;       6. CHECK FOR "A" INTERRUPT
5467                                    ;
5468                                    ;       7. CHECK FOR NO "B" INTERRUPT
5469
5470 033364                                    BGNSUB
      033364                                                                    19,4;
      033364  104402                                                                TRAP    C$BSUB
5471 033366  012737  177777  002274            MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
5472 033374  005037  002272                    CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5473 033400  112777  000000  146712            MOVB    #0,@#BSELO      ;DISABLE INTERRUPTS AGAIN
5474 033406  004537  004054                    JSR     R5,WRITEI       ;CAUSE AN INTERRUPT PENDING ON BOTH "A" & "B"
5475 033412  123005                            IRQREG
5476 033414  000000                            0
5477 033416  103003                            BCC     31$             ;IF AN ERROR OCCURED,
5478 033420                                    ERROR                   ;REPORT IT &
      033420  104460                                                                TRAP    C$ERROR
5479 033422                                    ESCAPE  TST             ;  QUIT
      033422  104410                                                                TRAP    C$ESCAPE
      033424  000414                                                                .WORD   L10051 .
5480
5481 033426  105737  002272            31$:    TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5482 033432  001407                            BEQ     10$             ;NO,  GOOD.  GO TEST THE "B" INTERRUPT
5483 033434  012737  000015  002254            MOV     #13.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5484 033442                                    GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                       ;       "DEVICE FATAL" ERROR # 64
      033442  104455                                                                TRAP    C$ERDF
      033444  000100                                                                .WORD   64
      033446  016526                                                                .WORD   EM34
      033450  006212                                                                .WORD   ERR1
5485
5486 033452  105737  002273            10$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5487 033456  001407                            BEQ     11$             ;NO,  GOOD.  NOW TRY LETTING ONE THROUGH
5488 033460  012737  000016  002254            MOV     #14.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5489 033466                                    GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                       ;       "DEVICE FATAL" ERROR # 65
      033466  104455                                                                TRAP    C$ERDF
      033470  000101                                                                .WORD   65
      033472  016540                                                                .WORD   EM34B
      033474  006212                                                                .WORD   ERR1
5490
5491 033476  005037  002272            11$:    CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5492 033502  112777  000021  146610            MOVB    #IENBA!IENBB,@#BSELO    ;ENABLE BOTH INTERRUPTS
5493 033510  012703  001000                    MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
5494 033514  077301                            SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
5495 033516  105737  002272                    TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5496 033522  001007                            BNE     12$             ;YES, GOOD.  GO TEST THE "B" INTERRUPT
5497 033524  012737  000017  002254            MOV     #15.,GDATA      ;NO,  TELL ERROR HANDLER WHAT WE HAD DONE
5498 033532                                    GEDF    EM35,ERR1       ;REPORT MISSING INTERRUPT ON "ENABLE"
                                                                       ;       "DEVICE FATAL" ERROR # 66
```

```
        033532  104455                                                          TRAP    C$ERDF
        033534  000102                                                          .WORD   66
        033536  016552                                                          .WORD   EM35
        033540  006212                                                          .WORD   ERR1
   5499
   5500 033542  105737  002273          12$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
   5501 033546  001407                  BEQ     13$             ;NO,  GOOD.  NOW TRY HITTING THE "B" INTERRUPT
   5502 033550  012737  000020  002254  MOV     #16.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
   5503 033556                          GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                ;       "DEVICE FATAL" ERROR # 67
        033556  104455                                                          TRAP    C$ERDF
        033560  000103                                                          .WORD   67
        033562  016540                                                          .WORD   EM34B
        033564  006212                                                          .WORD   ERR1
   5504
   5505 033566                  13$:    ENDSUB
        033566                                                          10055:
        033566  104403                                                          TRAP    C$ESUB
   5506
   5507                         ;-----------------------------------------------------------------------
   5508                         ;               TEST FOR "A" INTERRUPT WHILE ENABLED
   5509                         ;
   5510                         ;       1. ENABLE BOTH INTERRUPTS
   5511                         ;
   5512                         ;       2. ASSERT "B" REQUEST BIT TO 1:
   5513                         ;               DISABLING "B" & FORCING "A"
   5514                         ;
   5515                         ;       3. CHECK FOR "A" INTERRUPT
   5516                         ;
   5517                         ;       4. CHECK FOR NO "B" INTERRUPT
   5518
   5519 033570                  BGNSUB
        033570                                                          T9.5:
        033570  104402                                                          TRAP    C$BSUB
   5520 033572  012737  177777  002274          MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
   5521 033600  005037  002272          CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
   5522 033604  112777  000021  146506  MOVB    #IENBA!IENBB,@BSELO     ;ENABLE BOTH INTERRUPTS
   5523 033612  C04537  004054          JSR     R5,WRITEI       ;CAUSE AN INTERRUPT PENDING ON "A"
   5524 033616  123005                  IRQREG                  ;   BUT NOT ON "B"
   5525 033620  000002                  IRQB
   5526 033622  103003                  BCC     31$             ;IF AN ERROR OCCURED,
   5527 033624                          ERROR                   ;REPORT IT &
        033624  104460                                                          TRAP    C$ERROR
   5528 033626                          ESCAPE  TST             ;  QUIT
        033626  104410                                                          TRAP    C$ESCAPE
        033630  000210                                                          .WORD   10051
   5529
   5530 033632  012703  001000  31$:    MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
   5531 033636  077301                  SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
   5532 033640  105737  002272          TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
   5533 033644  001007                  BNE     12$             ;YES, GOOD.  GO TEST THE "B" INTERRUPT
   5534 033646  012737  000023  002254  MOV     #19.,GDATA      ;NO,  TELL ERROR HANDLER WHAT WE HAD DONE
   5535 033654                          GEDF    EM35,ERR1       ;REPORT MISSING INTERRUPT ON "ENABLE"
                                                                ;       "DEVICE FATAL" ERROR # 68
        033654  104455                                                          TRAP    C$ERDF
        033656  000104                                                          .WORD   68
        033660  016552                                                          .WORD   EM35
```

```
TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION

      033662  006212                                                                  .WORD   ERR1
5536
5537  033664  105737  002273          12$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
5538  033670  001407                  BEQ     13$             ;NO.  GOOD.  NOW TRY HITTING THE "B" INTERRUPT
5539  033672  012737  000024  002254          MOV     #20.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5540  033700                          GEDF    EM34B,ERR1      ;REPORT THE UNEXPECTED INTERRUPT
                                                                ;       "DEVICE FATAL" ERROR # 69
      033700  104455                                                                  TRAP    C$ERDF
      033702  000105                                                                  .WORD   69
      033704  016540                                                                  .WORD   EM34B
      033706  006212                                                                  .WORD   ERR1
5541
5542  033710                          13$:    ENDSUB
      033710                                                                  L10056:
      033710  104403                                                                  TRAP    C$ESUB
5543
5544                                  ;.....................................................................
5545                                  ;               TEST FOR "B" INTERRUPT WHILE ENABLED
5546                                  ;
5547                                  ;       1. ENABLE BOTH INTERRUPTS
5548                                  ;
5549                                  ;       2. ASSERT "A" REQUEST BIT TO 1:
5550                                  ;               DISABLING "A" & FORCING "B"
5551                                  ;
5552                                  ;       3. CHECK FOR NO "A" INTERRUPT
5553                                  ;
5554                                  ;       4. CHECK FOR "B" INTERRUPT
5555
5556  033712                          BGNSUB
      033712                                                                  T9.6:
      033712  104402                                                                  TRAP    C$BSUB
5557  033714  012737  177777  002274          MOV     #-1,INTWCH      ;TELL BOTH HANDLERS TO "WATCH" FOR INTERRUPTS
5558  033722  005037  002272                  CLR     INTFLG          ;CLEAR BOTH INTERRUPT FLAGS
5559  033726  112777  000021  146364          MOVB    #IENBA!IENBB,@BSELO     ;ENABLE BOTH INTERRUPTS
5560  033734  004537  004054                  JSR     R5,WRITEI       ;CAUSE AN INTERRUPT PENDING ON "B"
5561  033740  123005                          IRQREG                  ;   BUT NOT ON "A"
5562  033742  000004                          IRQA
5563  033744  103003                          BCC     32$             ;IF AN ERROR OCCURED,
5564  033746                                  ERROR                   ;REPORT IT &
      033746  104460                                                                  TRAP    C$ERROR
5565  033750                          ESCAPE  TST                     ;  QUIT
      033750  104410                                                                  TRAP    C$ESCAPE
      033752  000066                                                                  .WORD   L10051
5566
5567  033754  012703  001000          32$:    MOV     #1000,R3        ;GIVE THE INTERRUPT SOME TIME TO HAPPEN
5568  033760  077301                  SOB     R3,.            ;  BY SITTING HERE FOR A WHILE
5569  033762  105737  002272                  TSTB    INTFLG          ;DID AN "A" INTERRUPT OCCUR?
5570  033766  001407                  BEQ     16$             ;NO.  GOOD.  GO TEST THE "B" INTERRUPT
5571  033770  012737  000025  002254          MOV     #21.,GDATA      ;YES, TELL ERROR HANDLER WHAT WE HAD DONE
5572  033776                          GEDF    EM34,ERR1       ;REPORT THE UNEXPECTED INTERRUPT
                                                                ;       "DEVICE FATAL" ERROR # 70
      033776  104455                                                                  TRAP    C$ERDF
      034000  000106                                                                  .WORD   70
      034002  016526                                                                  .WORD   EM34
      034004  006212                                                                  .WORD   ERR1
5573
5574  034006  105737  002273          16$:    TSTB    INTFLG+1        ;DID A "B" INTERRUPT OCCUR?
```

TEST 9 -- Q-BUS INTERRUPT "A" & "B" SELECTION

```
5575 034012  001007                    BNE     17$             ;YES, GOOD.  NOW TRY HITTING THE "B" INTERRUPT
5576 034014  012737  000026  002254     MOV     #22.,GDATA      ;NO,  TELL ERROR HANDLER WHAT WE HAD DONE
5577 034022                            GEDF    EM35B,ERR1      ;REPORT MISSING INTERRUPT ON "ENABLE"
                                                               ;       "DEVICE FATAL" ERROR # 71
     034022  104455                                                                    TRAP    C$ERDF
     034024  000107                                                                    .WORD   71
     034026  016573                                                                    .WORD   EM35B
     034030  006212                                                                    .WORD   ERR1
5578
5579 034032                      17$:  ENDSUB
     034032                                                                L10057:
     034032  104403                                                                   TRAP    C$ESUB
5580
5581 034034  005037  002274            CLR     INTWCH          ;TELL HANDLERS TO STOP WATCHING FOR INTERRUPTS
5582 034040                            ENDTST
     034040                                                                L10051:
     034040  104401                                                                   TRAP    C$ETST
```

```
5591                                    .SBTTL TEST 10 -- BUS RESET WITH DISABLE INIT SET

                                        ;+*****************************************************************************
                                        ;*
                                        ;*      TEST 10 -- BUS RESET WITH DISABLE INIT SET
                                        ;*
                                        ;* A  BYTE SELECT REGISTER (BSEL3) IS LOADED WITH 377, DISABLE INIT BIT IS SET
                                        ;* IN  THE  NPR CONTROL REGISTER, AND A BUS RESET INSTRUCTION IS EXECUTED. THE
                                        ;* PROGRAM  THEN  CHECKS  THAT THE DMV-11 WAS NOT CLEARED, BY CHECKING FOR 377
                                        ;* STILL IN BSEL3
                                        ;*
                                        ;-*****************************************************************************
                                        ;
                                        ;       BGNTST
     034042                                                                                     T10::
5592 034042  032737  000001  002316            BIT     #BIT0,PFLAG     ;IF BUS RESETS ARE NOT ALLOWED,
5593 034050  001031                            BNE     10$             ;  BYPASS THIS TEST
5594                                                                   ;ELSE,
5595 034052  004737  003514                    JSR     PC,MSTCLR       ;INIT DMV & START UP THE MAINT. LOOP
5596 034056  103003                            BCC     1$              ;IF AN ERROR OCCURED,
5597 034060                                    ERROR                   ;REPORT IT &
     034060  104460                                                                     TRAP    C$ERROR
5598 034062                                    ESCAPE  TST             ;  EXIT
     034062  104410                                                                     TRAP    C$ESCAPE
     034064  000050                                                                     .WORD   L10060-.
5599
5600 034066  004537  004054            1$:     JSR     R5,WRITEI       ;NOW SET "DISABLE INIT"
5601 034072  123004                            NPRCTL
5602 034074  000105                            DMVDAI!DMVPU!NPRGO
5603                                                                   ;THE "NPRGO" BIT IS SET BECAUSE ASSERTING IT
5604                                                                   ;TO A ZERO WOULD KICK OFF AN NPR OPERATION!
5605                                                                   ;THE "DMVPU" BIT MUST ALWAYS BE SET WHENEVER
5606                                                                   ;THE NPR-CONTROL REGISTER IS LOADED.
5607
5608 034076  112777  000377  146222            MOVB    #377,@BSEL3     ;THIS REGISTER WILL ONLY GET ALTERED IF THE
5609                                                                   ;DMV-11 IS SUCCESSFULLY RESET:  THE "DMVPU"
5610                                                                   ;BIT WILL BE CLEARED, THE MICRO-DIAGNOSTIC
5611                                                                   ;WILL BE STARTED, AND FINDING "DMVPU" CLEARED,
5612                                                                   ;IT WILL CLEAR ALL BSEL REGISTERS (INCLUDING
5613                                                                   ;BSEL3) AND PERFORM THE 17 TESTS IS CONTAINS.
5614                                                                   ;OF COURSE, IF THIS ALL HAPPENS, THAN THIS
5615                                                                   ;TEST WILL HAVE FAILED!
5616
5617 034104                                    BRESET                  ;THE "SUPERVISOR" WILL DO A BUS RESET FOR US
     034104  104433                                                                     TRAP    C$RESET
5618 034106  012703  001000                    MOV     #1000,R3        ;DELAY FOR A BIT SO THE MICRO-DIAG. CAN DO
5619 034112  077301                            SOB     R3,.            ;ITS THING IF IT'S GOING TO
5620 034114  122777  000377  146204            CMPB    #377,@BSEL3     ;IF A FAILURE OCCURED, THIS SHOULD HAVE BEEN
5621 034122  001404                            BEQ     10$             ;ALTERED BY NOW.  IF NOT, ALL'S WELL -- EXIT
5622 034124                                    GEDF    EM40,ERR1       ;ELSE, "DISABL INIT" DIDN'T STOP "BUS RESET"
                                                                       ;       "DEVICE FATAL" ERROR # 72
     034124  104455                                                                     TRAP    C$EROF
     034126  000110                                                                     .WORD   72
     034130  016614                                                                     .WORD   EM40
     034132  006212                                                                     .WORD   ERR1
5623
5624 034134                            10$:    ENDTST
```

```
        034134
        034134  104401                          L10060:
                                                        TRAP    C$ETST
```

```
5635                                    .SBTTL TEST 11 -- MASTER CLEAR WITH DISABLE INIT SET

                                        ;+*****************************************************************************
                                        ;*
                                        ;*      TEST 11 -- MASTER CLEAR WITH DISABLE INIT SET
                                        ;*
                                        ;* THE "DISABL INIT" BIT IN THE NPR CONTROL REGISTER IS SET AND A MASTER CLEAR
                                        ;* IS  ISSUED.  IF  THE  MASTER  CLEAR SUBROUTINE DETECTS AN ERROR, THE MASTER
                                        ;* CLEAR  WILL  NOT  HAVE  FUNCTIONED PROPERLY. WHERE THE NORMAL ERROR MESSAGE
                                        ;* (QUEUED  UP  BY "MASCLR") IS NORMALLY PRINTED, THIS TEST WILL PRINT ITS OWN
                                        ;* INSTEAD.
                                        ;*
                                        ;--*****************************************************************************
                                        ;
                                        ;       BGNTST
     034136                                                                                  T11::
5636 034136  004737  003514                    JSR     PC,MSTCLR       ;INIT DMV & START UP THE MAINT. LOOP
5637 034142  103003                            BCC     1$              ;IF AN ERROR OCCURED,
5638 034144                                    ERROR                   ;REPORT IT &
     034144  104460                                                                     TRAP    C$ERROR
5639 034146                                    ESCAPE  TST             ;  EXIT
     034146  104410                                                                     TRAP    C$ESCAPE
     034150  000030                                                                     .WORD   L10061-.
5640 034152  004537  004054        1$:         JSR     R5,WRITEI       ;NOW SET "DISABLE INIT"
5641 034156  123004                            NPRCTL
5642 034160  000105                            DMVDAI!DMVPU!NPRGO
5643                                                                   ;THE "NPRGO" BIT IS SET BECAUSE ASSERTING IT
5644                                                                   ;TO A ZERO WOULD KICK OFF AN NPR OPERATION!
5645                                                                   ;THE "DMVPU" BIT MUST ALWAYS BE SET WHENEVER
5646                                                                   ;THE NPR-CONTROL REGISTER IS LOADED.
5647
5648 034162  004737  003514                    JSR     PC,MSTCLR       ;INIT THE DMV & RESTART THE M-LOOP
5649 034166  103004                            BCC     2$              ;IF AN ERROR OCCURED, IGNORE QUEUED ERROR AND
5650 034170                                    GEDF    EM41,ERR1       ;REPORT THAT "DISABL INIT" STOPPED MASTER CLEAR
                                        ;               "DEVICE FATAL" ERROR # 73
     034170  104455                                                                     TRAP    C$ERDF
     034172  000111                                                                     .WORD   73
     034174  016671                                                                     .WORD   EM41
     034176  006212                                                                     .WORD   ERR1
5651 034200                             2$:     ENDTST
     034200                                                                 L10061:
     034200  104401                                                                     TRAP    C$ETST
```

```
   5667                                 .SBTTL TEST 12 -- DCOK H LO BIT

                                        ;+*****************************************************************************
                                        ;*
                                        ;*      TEST 12 -- DCOK H LO BIT
                                        ;*
                                        ;* DCOK H LO IS SET IN THE NPR CONTROL REGISTER WHICH SHOULD CAUSE A VECTOR TO
                                        ;* THE  FIRST  INTERRUPT  HANDLER  WHERE THE VECTOR IS CHANGED TO POINT TO THE
                                        ;* SECOND HANDLER. THIS SECOND HANDLER WILL THEN STALL FOR A WHILE WAITING FOR
                                        ;* THE  POWER-UP  INTERRUPT  WHICH  SHOULD KICK US INTO THE SECOND HANDLER. IN
                                        ;* BOTH  HANDLERS  FLAGS  ARE  SET  TO  SAY THAT WE GOT THERE. WHEN WE FINALLY
                                        ;* RETURN  TO  OUR MAINLINE CODE, WE WILL RESUME THE DELAY FUNCTION WE WERE IN
                                        ;* AND THEN CHECK THE FLAGS.
                                        ;*
                                        ;* IN SUBTEST # 1, WE EXPECT THE DMV TO BE RESET.
                                        ;*
                                        ;*
                                        ;-*****************************************************************************
                                        ;
                                        ;       BGNTST
     034202                                                                          T12::
   5668          000014                 DCOKTS  = $T                    ;DEFINE TEST # FOR "INIT" SECTION
   5669 034202   032737  000001  002370         BIT     #PU24,PT.CTL    ;IS POWER-UP STRAPPED FOR OPTION 0?
   5670 034210   001002                        BNE     1$              ;YES, THEN WE CAN DO THIS TEST
   5671 034212                                 EXIT    TST             ;NO,  WE CAN'T DO THIS TEST UNLESS IT IS!
     034212   104432                                                                TRAP    C$EXIT
     034214   000476                                                                .WORD   L10062-.
   5672
   5673 034216   032737  000001  002316  1$:    BIT     #BIT0,PFLAG     ;IF BUS RESETS ARE ALLOWED,
   5674 034224   001402                        BEQ     2$              ;  PERFORM THIS TEST
   5675 034226                                 EXIT    TST             ;ELSE, BYPASS IT
     034226   104432                                                                TRAP    C$EXIT
     034230   000462                                                                .WORD   L10062-.
   5676
   5677 034232                          2$:    DELAY   40.             ;DELAY TO PREVENT TST # ROACHING
     034232   012727  000050                                                         MOV     #40.,(PC)+
     034236   000000                                                                .WORD   0
     034240   013727  002116                                                         MOV     L$DLY,(PC)+
     034244   000000                                                                .WORD   0
     034246   005367  177772                                                         DEC     -6(PC)
     034252   001375                                                                BNE     .-4
     034254   005367  177756                                                         DEC     -22(PC)
     034260   001367                                                                BNE     .-20
   5678                                 ;-----------------------------------------------------------------------------
   5679                                 ; SUBTEST #1: DCOK H LO (RESET DMV)
   5680                                 ;-----------------------------------------------------------------------------
   5681 034262                                 BGNSUB                  ; <*****>  TEST FOR POWER-DOWN/UP & DMV-11 RESET
     034262                                                                          T12.1:
     034262   104402                                                                TRAP    C$BSUB
   5682 034264   004737  003514                JSR     PC,MSTCLR       ;INIT DMV & START UP THE MAINT. LOOP
   5683 034270   103003                        BCC     3$              ;IF AN ERROR OCCURED,
   5684 034272                                 ERROR                   ;REPORT IT &
     034272   104460                                                                TRAP    C$ERROR
   5685 034274                                 ESCAPE  SUB             ;  EXIT
     034274   104410                                                                TRAP    C$ESCAPE
     034276   000220                                                                .WORD   L10063-.
   5686 034300                          3$:    SETVEC  #24,#5$,#7      ;SETUP VECTOR FOR POWER FAIL INTERRUPT HANDLER
```

TEST 12 -- DCOK H LO BIT

```
        034300  012746  000007                                                  MOV     #7,-(SP)
        034304  012746  034406                                                  MOV     #5$,-(SP)
        034310  012746  000024                                                  MOV     #24,-(SP)
        034314  012746  000003                                                  MOV     #3,-(SP)
        034320  104437                                                          TRAP    C$SVEC
        034322  062706  000010                                                  ADD     #10,SP
  5687  034326  112737  177777  002274          MOVB    #-1,INTWCH      ;EXPECT AN "A" INTERRUPT (IF "DCOK" FAILS!)
  5688  034334  105037  002272                  CLRB    INTFLG          ;CLEAR THE FLAG IN CASE WE WANT TO DETECT IT
  5689  034340  152777  000001  145752          BISB    #IENBA,@BSEL0   ;NOW ENABLE "A" INTERRUPTS
  5690
  5691  034346  010637  002454                  MOV     SP,OLDSP        ;SAVE THE STACK POINTER
  5692
  5693                                  ;       SETUP "DISABL INIT" TO ALLOW BINIT TO RESET THE DMV AND THEN CAUSE
  5694                                  ;       A POWER-FAIL CONDITION TO BE SIMULATED.
  5695
  5696  034352  012777  177506  145754          MOV     #177400!LSIDCL!DMVPU!NPRGO,@SEL6  ;VALUE TO BE LOADED INTO
  5697  034360  012777  123004  145742          MOV     #NPRCTL,@SEL4           ;THE NPR CONTROL REGISTER
  5698  034366  112777  000042  145730          MOVB    #WRILOC!<IRQA*8.>,@BSEL2  ;TELL M-LOOP TO WRITE IT & INTERRUPT
  5699                                                                          ;US JUST BEFORE REQUESTING ANOTHER
  5700                                                                          ;M-LOOP COMMAND.
  5701
  5702                                  ;       SETTING  "NPRGO" PREVENTS AN "NPR" OPERATION FROM OCCURING. THE HIGH
  5703                                  ;       ORDER BYTE IS SET TO -1 AND IS USED AS A FLAG -- WHEN THE RESET
  5704                                  ;       OCCURS, THE SELECT REGISTERS WILL ALL BE CLEARED; WE WILL BE
  5705                                  ;       LOOKING AT THE RUN BIT TO BE SET AND AT BSEL7 TO BE CLEARED AS POSITIVE
  5706                                  ;       PROOF THAT THE DMV GOT RESET.
  5707
  5708
  5709  034374  000001                  4$:     WAIT                    ;HANG HERE UNTIL INTERRUPTED
  5710
  5711
  5712                                  ;       IF THE DCOK WORKS AS IT SHOULD, THE NEXT INSTRUCTION TO BE EXECUTED
  5713                                  ;       IS AT THE LABEL "5$". IF I DOESN'T WORK, THE DMV SHOULD FINISH THE
  5714                                  ;       "WRITE" COMMAND AND GENERAT A Q-BUS INTERRUPT. ON RETURNING FROM
  5715                                  ;       THAT INTERRUPT, WE FALL INTO THE ERROR CALL BELOW:
  5716
  5717  034376                          GEDF    EM42A,ERR1      ;REPORT MISSING POWER-UP
                                                                ;       "DEVICE FATAL" ERROR # 74
        034376  104455                                                          TRAP    C$ERDF
        034400  000112                                                         .WORD   74
        034402  016746                                                         .WORD   EM42A
        034404  006212                                                         .WORD   ERR1
  5718
  5719  034406                          5$:
  5720
  5721                                  ;       IN EITHER CASE: RESTORE THE POWER FAIL VECTOR, STACK POINTER, AND
  5722                                  ;       THE FLAGS USED BY THE Q-BUS INTERRUPT SERVICE ROUTINE.
  5723
  5724                                  ;                               B U T   F I R S T !
  5725
  5726                                  ;       IN SOME CASES THE Q-BUS GETS CONFUSED WHEN WE PERFORM THE ABOVE
  5727                                  ;       (HIGHLY NON- STANDARD) "DCOK" MANIPULATION. EXPERIENCE HAS SHOWN
  5728                                  ;       THAT THE INSTRUCTION BEING EXECUTED CAN BE CORRUPTED -- USUALLY (BUT
  5729                                  ;       NOT ALWAYS) BEING CLEARED TO ZERO (A HALT INSTRUCTION). THIS IS NOT
  5730                                  ;       A FAILURE OF THE DMV AND THEREFORE SHOULD NOT BE OUR CONCERN HERE.
  5731                                  ;       THE FAILURE SHOULD NOT AFFECT THE TEST WITHIN THE PASS IN WHICH IT
  5732                                  ;       OCCURS BUT IN A SUBSEQUENT PASS. IN AN EFFORT TO ELIMINATE ANY
```

TEST 12 -- DCOK H LO BIT

```
5733                                    ;       PROBLEMS FROM THIS GLITCH, WE RESTORE THE INSTRUCTION:
5734
5735 034406  012737  000001  034374      MOV     #1,4$           ;RESTORE THE "WAIT" INSTRUCTION -- JUST IN
5736                                                            ; CASE IT GOT MODIFIED!
5737 034414  013706  002454              MOV     OLDSP,SP        ;RESTORE THE STACK
5738 034420  142777  000001  145672      BICB    #IENBA,@BSELO   ;DISABLE THE "A" INTERRUPT
5739 034426  105037  002274              CLRB    INTWCH          ;STOP EXPECTING Q-BUS INTERRUPTS
5740 034432                              CLRVEC  #24             ;RETURN THE VECTOR TO THE SUPERVISOR
     034432  012700  000024                                                      MOV     #24,R0
     034436  104436                                                              TRAP    C$CVEC
5741 034440                              SETPRI  #0              ;MAKE SURE WE'RE BACK RUNNING AT 0 AGAIN!
     034440  012700  000000                                                      MOV     #0,R0
     034444  104441                                                              TRAP    C$SPRI
5742
5743 034446  013701  002262              MOV     DELAY1,R1       ;INITIALIZE TTHE LOOP COUNTER FOR DELAY LOOP
5744 034452  001402          10$:        BEQ     11$             ; EXIT DELAY LOOP IF TIME HAS EXPIRED
5745 034454  005301                      DEC     R1              ;  ELSE, DECREMENT THE LOOP COUNTER AND
5746 034456  000775                      BR      10$             ;  CONTINUE TO LOOP
5747 034460                      11$:                            ; TIME UP !
5748 034460  132777  000200  145634      BITB    #RUN,@BSEL1     ;CHECK RUN BIT
5749 034466  001403                      BEQ     12$             ; NOT SET... REPORT ERROR.
5750 034470  105777  145642              TSTB    @BSEL7          ;THIS REGISTER SHOULD HAVE BEEN CLEARED
5751 034474  001404                      BEQ     13$             ;IT IS, EVERYTHING HERE IS OK -- EXIT SUBTEST
5752 034476                      12$:    GEDF    EM42B,ERR1      ;NO, THEN REPORT THE FAILURE
                                                                 ;       "DEVICE FATAL" ERROR # 75
     034476  104455                                                              TRAP    C$ERDF
     034500  000113                                                              .WORD   75
     034502  017020                                                              .WORD   EM42B
     034504  006212                                                              .WORD   ERR1
5753 034506  004737  003346      13$:    JSR     PC,MASCLR       ;RESTORE DMV-11 TO A NORMAL STATE!
5754 034512  103001                      BCC     14$             ;NO ERRORS, EXIT SUBTEST
5755 034514                              ERROR                   ;REPORT MSTCLR ERROR
     034514  104460                                                              TRAP    C$ERROR
5756 034516                      14$:
5757 034516                              ENDSUB
     034516                                                                  L10063:
     034516  104403                                                              TRAP    C$ESUB
5758                                    ;-------------------------------------------------------------------------
5759                                    ; SUBTEST #2: DCOK H LO (DMV-11 SHOULDN'T BE RESET)
5760                                    ;-------------------------------------------------------------------------
5761                                    ;       SINCE HITTING "DCOK H LO" WITHOUT "HALT" OCCASIONALLY CORRUPTS
5762                                    ;       PROGRAM MEMORY: WE SET BOTH "HALT" AND "DCOK H LO" IN THIS
5763                                    ;       SUB-TEST (IF HALT FAILS, THIS TEST MAY BLOW UP).
5764
5765 034520                              BGNSUB          ; <***> TEST FOR POWER-DOWN/UP & NO DMV-11 RESET
     034520                                                                  T12.2:
     034520  104402                                                              TRAP    C$BSUB
5766
5767 034522  004737  003514              JSR     PC,MSTCLR       ;INIT DMV & START UP THE MAINT. LOOP
5768 034526  103003                      BCC     2$              ;IF AN ERROR OCCURED,
5769 034530                              ERROR                   ;REPORT IT &
     034530  104460                                                              TRAP    C$ERROR
5770 034532                              ESCAPE  SUB             ;  EXIT
     034532  104410                                                              TRAP    C$ESCAPE
     034534  000154                                                              .WORD   L10064
5771 034536                      2$:     SETVEC  #24,#5$,#7      ;SETUP VECTOR FOR POWER FAIL INTERRUPT HANDLER
     034536  012746  000007                                                      MOV     #7,-(SP)
```

TEST 12 -- DCOK H LO BIT

```
        034542  012746  034640                                                  MOV     #5$,-(SP)
        034546  012746  000024                                                  MOV     #24,-(SP)
        034552  012746  000003                                                  MOV     #3,-(SP)
        034556  104437                                                          TRAP    C$SVEC
        034560  062706  000010                                                  ADD     #10,SP
  5772  034564  010637  002454            MOV     SP,OLDSP        ;SAVE THE STACK POINTER
  5773
  5774  034570  004537  004166            JSR     R5,MOVLTD       ;MOVE THE MICRO CODE INTO THE DMV
  5775  034574  034714                    SMCODE                  ;       THIS IS WHERE IT STARTS
  5776  034576  000040                    EMCODE-SMCODE           ;       THIS IS ITS SIZE IN BYTES
  5777
  5778  034600  112777  000377  145526    MOVB    #377,@BSEL6     ;WRITE ALL 1'S TO BSEL6
  5779
  5780  034606  012777  000077  145514    MOV     #77,@SEL4       ;START ADDRESS OF MICROCODE
  5781  034614  012777  000005  145502    MOV     #EXECUT,@SEL2   ;INITIATE M-CODE
  5782
  5783                          ;  *** IF THE RESET GETS THROUGH, THE MICRO-DIAGNOSTIC WILL CLEAR BSEL4 ***
  5784
  5785  034622  012703  001000            MOV     #1000,R3        ;STALL FOR A BIT (UCODE SHOULD HALT US HERE)
  5786  034626  077301                    SOB     R3,.
  5787
  5788                          ; IF WE GET HERE, WE NEVER GOT THE EXPECTED POWER-UP SEQUENCE!
  5789
  5790  034630                            GEDF    EM42A,ERR1      ;REPORT MISSING POWER-UP
                                                                    ;       "DEVICE FATAL" ERROR # 76
        034630  104455                                                          TRAP    C$ERDF
        034632  000114                                                          .WORD   76
        034634  016746                                                          .WORD   EM42A
        034636  006212                                                          .WORD   ERR1
  5791
  5792                          ;IN EITHER CASE, RESTORE THE VECTOR & STACK AND SEE IF THE DMV GOT RESET
  5793
  5794  034640  013706  002454    5$:     MOV     OLDSP,SP        ;RESTORE THE STACK
  5795  034644                            CLRVEC  #24             ;RETURN THE VECTOR TO THE SUPERVISOR
        034644  012700  000024                                                  MOV     #24,R0
        034650  104436                                                          TRAP    C$CVEC
  5796  034652                            SETPRI  #0              ;MAKE SURE WE'RE BACK RUNNING AT 0 AGAIN!
        034652  012700  000000                                                  MOV     #0,R0
        034656  104441                                                          TRAP    C$SPRI
  5797  034660  122777  000377  145446    CMPB    #377,@BSEL6     ;THIS REGISTER SHOULD NOT HAVE BEEN CLEARED
  5798  034666  001404                    BEQ     10$             ;IT ISN'T, ALL IS OK -- EXIT SUBTEST
  5799  034670                            GEDF    EM42C,ERR1      ;IT IS, THEN REPORT THE FAILURE
                                                                    ;       "DEVICE FATAL" ERROR # 77
        034670  104455                                                          TRAP    C$ERDF
        034672  000115                                                          .WORD   77
        034674  017074                                                          .WORD   EM42C
        034676  006212                                                          .WORD   ERR1
  5800                                                              ;* AT THIS POINT "DINIT" IS STILL SET *
  5801  034700  004737  003346    10$:    JSR     PC,MASCLR       ;MAKE SURE THE DMV IS PROPERLY RESET!
  5802  034704  103001                    BCC     11$             ;EVERYTHING OK, EXIT SUBR AND TEST.
  5803  034706                            ERROR                   ;REPORT MASCLR ERROR
        034706  104460                                                          TRAP    C$ERROR
  5804  034710                    11$:
  5805  034710                            ENDSUB
        034710                                                              L10064:
        034710  104403                                                          TRAP    C$ESUB
  5806  034712                            ENDTST
```

TEST 12 -- DCOK H LO BIT

```
        034712
        034712  104401                  L10062:
                                                TRAP    C$ETST
```

```
                                        .SBTTL  SUBTEST 02'S M CODE  COMPLETE
                                        ;
                                        ; 6502 MICROCODE FOR TEST 0117 SUBTEST 02
                                        ;
                                        SMCODE:
                                                .BYTE 251,125   ;A9 55     LDA @GOBSY1!HALT!PWRUP!DINIT
                                                .BYTE 215,4,246 ;8D 04 A6  STA NPRCTL          ;SET DISABLE INIT/HALT
                                                .BYTE 251,105   ;A9 45     LDA @GOBSY1!PWRUP!DINIT
                                                .BYTE 215,4,246 ;8D 04 A6  STA NPRCTL          ;CLEAR HALT
                                                .BYTE 352       ;EA        NOP                 ;WAIT A WHILE
                                                .BYTE 352       ;EA        NOP
                                                .BYTE 352       ;EA        NOP
                                                .BYTE 251,107   ;A9 47     LDA @GOBSY1!PWRUP!SDCOK!DINIT
                                                .BYTE 215,4,246 ;8D 04 A6  STA NPRCTL          ;SET DCOK
                                                .BYTE 251,105   ;A9 45     LDA @GOBSY1!PWRUP!DINIT
                                                .BYTE 215,4,246 ;8D 04 A6  STA NPRCTL          ;CLEAR DCOK
                                                .BYTE 251,3     ;A9 03     LDA #3              ;DELAY FOR 16.8 USEC
                                                .BYTE 205,0     ;85 00     STA SP0             ;BUS INIT IS 10 USEC
                                                .BYTE 306,0     ;C6 00  5$:DEC SP0             ;
                                                .BYTE 320,374   ;D0 FC     BNE 5$              ;
                                                .BYTE 140       ;60        RTS          ;RETURN TO M LOOP
                                                .EVEN
                                        EMCODE:
```

```
5996            .SBTTL TEST 13    HALT MODE VERIFICATION

                ;;*********************************************************************
                ;*
                ;*      TEST 13 - HALT MODE VERIFICATION
                ;*
                ;* THIS TEST CONTAINS TWO (2) SUBTESTS DESIGNED TO VERIFY THE FUNCTIONALITY
                ;* OF THE "HALT" CONTROL CONTAINED WITHIN THE NPR CONTROL REGISTER. IN EACH
                ;* CASE, MICROCODE IS LOADED INTO THE DMV IN ORDER TO CONTROL THE TESTING
                ;* FROM THERE.
                ;*
                ;*---------------------------------------------------------------------
                ;*
                ;* SUBTEST # 1:
                ;*
                ;* HERE WE VERIFY THAT WE CAN CONTROL NPR'S AND DCOK PROPERLY WHILE THE 11 CPU
                ;* IS HALTED.
                ;*
                ;*      11 CPU'S OPERATIONS:                      DMV 11'S OPERATIONS:
                ;*
                ;* THE MICROCODE IS MOVED INTO THE DMV.
                ;*
                ;* CLEAR TMPO.  THIS WILL BE OUR TEST
                ;* LOCATION FOR THE NPR OPERATION.
                ;*
                ;* SETUP FOR POWER-FAIL VECTORING THROUGH
                ;* LOCATION 24.
                ;*
                ;* THE MICROCODE IS INITIATED & BSEL7 IS
                ;* SET TO -1 AS A FLAG.
                ;*
                ;* WAIT FOR BSEL7 TO BE CLEARED            CLEAR BSEL7 AND WAIT FOR IT TO GO
                ;*                                         NON ZERO AGAIN.  THIS PUTS THE
                ;*                                         DMV IN SYNC. WITH THE 11 CPU
                ;*
                ;* SAVE R6 IN OLDSP FOR RECOVERY LATER
                ;* CLEAR TMPO, LOAD INTO SEL4 THE
                ;* ADDRESS OF TMPO, AND SET BSEL7 TO -1.
                ;*
                ;* START LOOPING -- INCREMENTING TMPO       GET THE ADDRESS OF TMPO FROM SEL6
                ;*                                         AND SAVE IT FOR LATER
                ;*
                ;*                                         HALT THE 11 CPU.
                ;*
                ;* CONSOLE "ODT" SHOULD BE ENTERED.        NPR-IN THE CURRENT CONTENTS OF TMPO
                ;*                                         & PUT IT INTO SEL4 (THE FULL WORD).
                ;*
                ;*                                         DELAY FOR ABOUT 100 MICROSECONDS
                ;*                                         (THE TIME ISN'T CRITICAL).
                ;*
                ;* THE 11 CPU SHOULD NOT BE EXECUTING
                ;* ANYTHING NOW -- NOT EVEN "ODT"
                ;*
                ;*                                         DROP THE "HALT" SIGNAL TO RELEASE
                ;*                                         THE 11 CPU AND SET "DCOK H LO" &
                ;*                                         "DISABL INIT". DROP "DCOK H LO"
                ;*
```

```
;*
;* WE SHOULD GO
;* THROUGH A POWER-UP SEQUENCE.   R6 IS
;* RESTORED FROM OLDSP, INTERRUPT
;* PRIORITY LEVEL IS RESTORED TO 0, &
;* INTERRUPT VECTOR 24 IS RETURNED TO
;* THE DIAGNOSTIC SUPERVISOR.  SEL4 IS
;* COMPARED AGAINST TMP0 -- THEY SHOULD
;* BE EQUAL.
;*
;* NOW CLEAR BSEL7.                        WAIT FOR BSEL7 TO CLEAR, THEN DROP
;*                                         "DISABLE INIT" AND EXIT TO M-LOOP.
;*
;*-------------------------------------------------------------------------
;*
;* SUBTEST # 2:
;*
;* HERE  THE  HALT MODE IS USED IN A WAY WHICH VERY CLOSELY MATCHES THE DMV-11
;* MICROCODE'S  UTILIZATION  DURING  A  "MOP  BOOT"  OPERATION. THE INTERRUPT
;* VECTOR  AREA  IS COMPLETELY OVERWRITTEN BY THE DMV NPR'S AND IS THEREFORE
;* BACKED  UP  ELSEWHERE  IN  THE  11 CPU'S MEMORY. THERE IS ALSO THE POSSIBLE
;* CONTENTION  WITH  THE  DIAGNOSTIC  SUPERVISOR  --  TO HELP HERE, AS MUCH AS
;* POSSIBLE WILL BE DONE AT INTERRUPT LEVEL 7.
;*
;*      11 CPU'S OPERATIONS:                    DMV-11'S OPERATIONS:
;*
;* THE MICROCODE IS MOVED INTO THE DMV.
;*
;* THE INTERRUPT VECTOR AREA IS BACKED-
;* UP IN AN I/O BUFFER FOLLOWING THE
;* PROGRAM
;*
;* THE MICROCODE IS INITIATED & BSEL7 IS
;* SET TO -1 AS A FLAG.
;*
;* WAIT FOR BSEL7 TO BE CLEARED            CLEAR BSEL7 AND WAIT FOR IT TO GO
;*                                         NON-ZERO AGAIN PUTTING BOTH
;*                                         PROCESSORS IN SYNC. WITH EACH
;*                                         OTHER
;*
;* CLEAR TMP0 AND SAVE R6 FOR RECOVERY
;* LATER.  SET BSEL7 AGAIN AND WAIT FOR
;* TMP0 TO BE SET.
;*                                         SET HALT, "DCOK H LO", & "DISABL
;*                                         INIT" AND PERFORM 2 NOP'S AS A
;*                                         1 MICROSECOND DELAY
;*
;* ENTRY INTO THE CONSOLE "ODT" WILL       CLEAR "DCOK H LO", SET "HALT" &
;* BE INITIATED.                           "DISABL INIT"
;*
;*                                         NPR OUT THE FOLLOWING:
;*                                          LOC:  CONTENTS
;*                                           24   000000  VECTOR TO LOC 0
;*                                           26   000340  @ PRIORITY 7
;*                                            0   012700  MOV  # 1,R0
;*                                            2   177777
;*                                            4   000777  BR   .
;*                                         ------ ------ -----------
;*
```

```
                ;*
                ;* THE ENTRY INTO THE CONSOLE "ODT"
                ;* WILL BE ABORTED!
                ;*
                ;*
                ;*
                ;*
                ;* THE ROUTINE NPR'D INTO LOC. 0 WILL
                ;* BE EXECUTED -- EVENTUALLY HANGING
                ;* AT THE "BR  ." INSTRUCTION @ LOC. 4.
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;*
                ;* THE ROUTINE JUST LOADED BY THE DMV
                ;* MICROCODE WILL NOW BE EXECUTED (WE
                ;* HOPE).  WHEN THE SUBROUTINE IS RE-
                ;* ENTERED (@ HLTST2),
                ;*
                ;*    1  THE INTERRUPT VECTORS WILL BE
                ;*       RESTORED;
                ;*    2  THE PRIORITY LEVEL WILL BE
                ;*       LOWERED BACK TO 0;
                ;*    3  R0, R1, & R6 WILL ALL BE CHECKED
                ;*       FOR THE PROPER CONTENTS; AND
                ;*    4  TMP0 WILL BE CHECKED FOR THE
                ;*       PROPER CONTENTS;
                ;*
                ;* ***************************************************************************

                        BGNTST

034/54
5997            000015  HLTEST  - $T                    ;DEFINE TEST 0 FOR "INIT" SECTION
                                                                                T13::
```

SET "DCOK H LO" & "DISABL INIT"
AND CLEAR "HALT"

DELAY FOR A SHORT TIME (ABOUT 1
MICROSECOND

CLEAR "DCOK H LO" AND SET "DISABL
INIT"

NPR-OUT THE FOLLOWING:

LOC:	CONTENTS		
6	005001	CLR	R1
10	062701	ADD	#(.+2),R1
12	062701	ADD	#(.+2),R1
.	.	.	
.	.	.	
.	.	.	
360	062701	ADD	#(.+2),R1
362	062701	ADD	#(.+2),R1
364	010037	MOV	R0,@#TMP0
366	(TMP0)		
370	013706	MOV	@#OLDSP,R6
372	(OLDSP)		
374	000137	JMP	HLTST2
376	(HLTST2)		

THIS IS SYNONYMOUS TO THE DMV-11
LOADING A "MESSAGE" STARTING AT MEM.
LOC. 000006.

NPR-OUT THE FOLLOWING:

LOC:	CONTENTS		
4	000240	NOP	

THIS IS HOW THE DMV-11 WILL TAKE THE
11 CPU OUT OF THE "BR ." CONDITION.

DROP "DISABL INIT"

AN EXIT IS TAKEN TO THE M-LOOP

TEST 13 -- HALT MODE VERIFICATION

```
 5998 034754  032737  000001  002370          BIT     #PU24,PT.CTL    ;IS POWER-UP STRAPPED FOR OPTION 0?
 5999 034762  001002                          BNE     5$              ;YES, THEN WE CAN DO THIS TEST
 6000 034764                                  EXIT    TST             ;NO,  WE CAN'T DO THIS TEST UNLESS IT IS!
      034764  104432                                                                  TRAP    C$EXIT
      034766  000772                                                                  .WORD   L10065-.
 6001
 6002 034770  032737  000001  002316  5$:     BIT     #BIT0,PFLAG     ;IF BUS RESETS ARE ALLOWED,
 6003 034776  001402                          BEQ     2$              ;  PERFORM THIS TEST
 6004 035000                                  EXIT    TST             ;ELSE, BYPASS IT
      035000  104432                                                                  TRAP    C$EXIT
      035002  000756                                                                  .WORD   L10065 .
 6005
 6006 035004                          2$:     DELAY   40.             ;DELAY TO PREVENT TST 0 ROACHING
      035004  012727  000050                                                          MOV     #40.,(PC)+
      035010  000000                                                                  .WORD   0
      035012  013727  002116                                                          MOV     L$DLY,(PC)+
      035016  000000                                                                  .WORD   0
      035020  005367  177772                                                          DEC     -6(PC)
      035024  001375                                                                  BNE     .-4
      035026  005367  177756                                                          DEC     -22(PC)
      035032  001367                                                                  BNE     .-20
 6007                                 ;-------------------------------------------------------------------
 6008 035034                          1$:     BGNSUB
      035034                                                                  T13.1:
      035034  104402                                                                  TRAP    C$BSUB
 6009 035036  004737  003514                  JSR     PC,MSTCLR       ;RESET DMV & ENTER M-LOOP
 6010 035042  103003                          BCC     2$              ;IF NO ERROR HERE, CONTINUE
 6011 035044                                  ERROR                   ;ELSE, REPORT THE ERROR
      035044  104460                                                                  TRAP    C$ERROR
 6012 035046                                  ESCAPE  TST             ;       & EXIT THE TEST
      035046  104410                                                                  TRAP    C$ESCAPE
      035050  000710                                                                  .WORD   L10065-.
 6013 035052                          2$:
 6014 035052  004537  004166                  JSR     R5,MOVLTD       ;MOVE THE MICRO CODE INTO THE DMV
 6015 035056  035762                          MC1                     ;       THIS IS WHERE IT STARTS
 6016 035060  000114                          MC2-MC1                 ;       THIS IS ITS SIZE IN BYTES
 6017
 6018 035062  005037  002412                  CLR     TMP0            ;INITIALIZE THE COUNTER
 6019 035066                                  SETVEC  #24,#24$,#7     ;SETUP POWER UP VECTOR
      035066  012746  000007                                                          MOV     #7,-(SP)
      035072  012746  035242                                                          MOV     #24$,-(SP)
      035076  012746  000024                                                          MOV     #24,-(SP)
      035102  012746  000003                                                          MOV     #3,-(SP)
      035106  104437                                                                  TRAP    C$SVEC
      035110  062706  000010                                                          ADD     #10,SP
 6020 035114  012777  000077  145206          MOV     #77,@SEL4       ;START ADDRESS OF MICROCODE
 6021 035122  112777  177777  145206          MOVB    #-1,@BSEL7      ;SET FLAG (BSEL7)
 6022 035130  012777  000005  145166          MOV     #EXECUT,@SEL2   ;INITIATE M-CODE
 6023
 6024 035136  005002                          CLR     R2
 6025 035140  105777  145172          3$:     TSTB    @BSEL7          ;WAIT FOR FLAG TO BE CLEARED
 6026 035144  001405                          BEQ     5$
 6027 035146  077204                          SOB     R2,3$
 6028 035150                                  GEDF    EM43A,ERR1      ;TIMEOUT...M CODE IS HUNG!
                                              ;       "DEVICE FATAL" ERROR # 28
      035150  104455                                                                  TRAP    C$ERDF
      035152  000116                                                                  .WORD   78
```

TEST 13 - HALT MODE VERIFICATION

```
                035154  017153                                                          .WORD   EM43A
                035156  006212                                                          .WORD   ERR1
        6029
        6030    035160  010637  002454          5$:     MOV     SP,OLDSP        ;SAVE STACK POINTER FOR LATER
        6031    035164  012777  002412  145136          MOV     #TMP0,@SEL4     ;PASS ADDRESS OF TMP0 TO M-CODE
        6032    035172  112777  177777  145136          MOVB    #1,@SEL7        ;TELL M-CODE TO PROCEED
        6033
        6034    035200  005237  002412          4$:     INC     TMP0            ;LOOP HERE UNTIL TMP0 GOES TO 0 AGAIN
        6035    035204  001375                          BNE     4$
        6036    035206  000240                          NOP
        6037    035210  000240                          NOP
        6038    035212  000240                          NOP
        6039    035214                                  GEDF    EM43A,ERR1      ;DMV SEEMS TO HAVE HUNG!
                                                                                ;       "DEVICE FATAL" ERROR # 79
                035214  104455                                                          TRAP    C$ERDF
                035216  000117                                                          .WORD   79
                035220  017153                                                          .WORD   EM43A
                035222  006212                                                          .WORD   ERR1
        6040    035224  000240                          NOP                     ; (FOR PATCHING)
        6041    035226  004737  003346                  JSR     PC,MASCLR       ;RESET THE DMV
        6042    035232  103001                          BCC     9$              ;RESET SUCCEEDED, ESCAPE TEST
        6043    035234                                  ERROR                   ; REPORT RESET ERROR
                035234  104460                                                          TRAP    C$ERROR
        6044    035236                          9$:     ESCAPE  TST             ; & GET OUT
                035236  104410                                                          TRAP    C$ESCAPE
                035240  000520                                                          .WORD   L10065-.
        6045
        6046    035242  013706  002454          24$:    MOV     OLDSP,SP        ;RESTORE R6 FIRST!
        6047    035246                                  CLRVEC  #24             ;RETURN THE VECTORE TO THE SUPERVISOR
                035246  012700  000024                                                  MOV     #24,R0
                035252  104436                                                          TRAP    C$CVEC
        6048    035254                                  SETPRI  #0              ;RESTORE PRIORITY LEVEL TO 0
                035254  012700  000000                                                  MOV     #0,R0
                035260  104441                                                          TRAP    C$SPRI
        6049    035262  027737  145042  002412          CMP     @SEL4,TMP0      ;THESE SHOULD BE EQUAL
        6050    035270  001405                          BEQ     30$             ;THEY ARE -- TEST PASSED
        6051    035272                                  GEDF    EM438,ERR1      ;THEY AREN'T -- HALT DIDN'T WORK
                                                                                ;       "DEVICE FATAL" ERROR # 80
                035272  104455                                                          TRAP    C$ERDF
                035274  000120                                                          .WORD   80
                035276  017177                                                          .WORD   EM43B
                035300  006212                                                          .WORD   ERR1
        6052    035302  000240                          NOP                     ; (FOR PATCHING)
        6053    035304  105077  145026          30$:    CLRB    @RSEL7          ;TELL M-CODE TO CLEAR DINIT
        6054    035310                                  ENDSUB
                035310                                                                  L10066:
                035310  104403                                                          TRAP    C$ESUB
        6055                                    ;-----------------------------------------------------------------
        6056    035312                                  BGNSUB
                035312                                                                  T13.2:
                035312  104402                                                          TRAP    C$BGSUB
        6057    035314  004737  003514                  JSR     PC,MSTCLR       ;RESET DMV & ENTER M-LOOP
        6058    035320  103003                          BCC     1$              ;IF NO ERROR HERE, CONTINUE
        6059    035322                                  ERROR                   ;ELSE, REPORT THE ERROR
                035322  104460                                                          TRAP    C$ERROR
        6060    035324                                  ESCAPE  TST             ;       & EXIT THE TEST
                035324  104410                                                          TRAP    C$ESCAPE
```

TEST 13 -- HALT MODE VERIFICATION

```
      035326  000432                                                                .WORD   L10065-.
 6061 035330                          1$:
 6062 035330  004537  004166                JSR     R5,MOVLTD       ;MOVE THE MICRO CODE INTO THE DMV
 6063 035334  036076                        MC2                     ;       THIS IS WHERE IT STARTS
 6064 035336  000243                        MC2END-MC2              ;       THIS IS ITS SIZE IN BYTES
 6065
 6066 035340  004537  005004                JSR     R5,MOVSW        ;SAVE THE INTERRUPT VECTORS
 6067 035344  000000                        0
 6068 035346  002654                        BUFAREA                 ; IN THE BUFFER AREA
 6069 035350  000200                        400/2                   ;  LOC'S 0 ==> 377 WILL BE SAVED
 6070
 6071 035352                                SETVEC  #376,#0,#7      ;FAKE OUT THE SUPERVISOR,  WE'RE JUST
      035352  012746  000007                                                        MOV     #7,-(SP)
      035356  012746  000000                                                        MOV     #0,-(SP)
      035362  012746  000376                                                        MOV     #376,(SP)
      035366  012746  000003                                                        MOV     #3,-(SP)
      035372  104437                                                               TRAP    C$SVEC
      035374  062706  000010                                                        ADD     #10,SP
 6072                                                           ; SETTING LOCATION 376 TO ZERO (0)
 6073
 6074 035400  012777  000077  144722        MOV     #77,@SEL4       ;START ADDRESS OF MICROCODE
 6075 035406  112777  177777  144722        MOVB    #-1,@BSEL7      ;SET FLAG (BSEL7)
 6076 035414  012777  000005  144702        MOV     #EXECUT,@SEL2   ;INITIATE M-CODE
 6077
 6078 035422  005002                        CLR     R2
 6079 035424  105777  144706          3$:   TSTB    @BSEL7          ;WAIT FOR FLAG TO BE CLEARED
 6080 035430  001406                        BEQ     5$
 6081 035432  077204                        SOB     R2,3$
 6082 035434                                GEDF    EM43A,ERR1      ;TIMEOUT...M-CODE IS HUNG!
                                                                    ;       "DEVICE FATAL" ERROR # 81
      035434  104455                                                               TRAP    C$ERDF
      035436  000121                                                               .WORD   81
      035440  017153                                                               .WORD   EM43A
      035442  006212                                                               .WORD   ERR1
 6083 035444  000447                        BR      22$             ;EXIT
 6084
 6085 035446  010637  002454          5$:   MOV     SP,OLDSP        ;SAVE STACK POINTER FOR LATER
 6086 035452  005037  002412                CLR     TMP0            ;RESET EXECUTION INDICATOR (TMP0)
 6087 035456  112777  177777  144652        MOVB    #-1,@BSEL7      ;TELL M-CODE TO PROCEED
 6088
 6089 035464  005003                        CLR     R3              ;WE'LL WAIT THIS LONG FOR THE M-CODE TO
 6090                                                               ;       INTERRUPT OUR SEQUENCE OF OPERATION
 6091 035466  005737  000376          10$:  TST     @#376           ;LOOK FOR THE M-CODE TO LOAD THIS LOCATION
 6092 035472  001017                        BNE     20$             ;WE SHOULD NEVER SEE THIS HAPPEN!!!
 6093 035474  077304                        SOB     R3,10$          ;LOOP UNTIL WE'RE INTERRUPTED
 6094                                                               ;IF WE AREN'T, WE HAVE A REAL PROBLEM
 6095
 6096 035476  004537  005004                JSR     R5,MOVSW        ;RESTORE THE INTERRUPT VECTORS
 6097 035502  002654                        BUFAREA                 ; FROM THE BUFFER AREA
 6098 035504  000000                        0
 6099 035506  000200                        400/2                   ; TO LOC'S 0 ==> 377
 6100
 6101 035510                                SETPRI  #0              ;RESTORE PRIORITY LEVEL TO 0
      035510  012700  000000                                                        MOV     #0,R0
      035514  104441                                                               TRAP    C$SPRI
 6102
 6103 035516                                GEDF    EM43A,ERR1      ;BUT WE AREN'T SURE WHAT IT IS!!!
```

```
                                                                ;       "DEVICE FATAL" ERROR # 82
     035516  104455                                                                       TRAP    C$ERDF
     035520  000122                                                                       .WORD   82
     035522  017153                                                                       .WORD   EM43A
     035524  006212                                                                       .WORD   ERR1
6104 035526  000240                        NOP                 ; (FOR PATCHING)
6105 035530  000415                        BR      22$         ; GO RESET DMV & EXIT
6106
6107 035532  004537  005004      20$:      JSR     R5,MOVSW    ;RESTORE THE INTERRUPT VECTORS
6108 035536  002654                        BUFAREA             ; FROM THE BUFFER AREA
6109 035540  000000                        0
6110 035542  000200                        400/2               ; TO LOC'S 0 ==> 377
6111
6112 035544                                SETPRI  #0          ;RESTORE PRIORITY LEVEL TO 0
     035544  012700  000000                                                             MOV     #0,R0
     035550  104441                                                                     TRAP    C$SPRI
6113
6114 035552                                GEDF    EM43C,ERR1  ;POWER-UP INTERRUPT DIDN'T OCCUR
                                                                ;       "DEVICE FATAL" ERROR # 83
     035552  104455                                                                     TRAP    C$ERDF
     035554  000123                                                                     .WORD   83
     035556  017215                                                                     .WORD   EM43C
     035560  006212                                                                     .WORD   ERR1
6115 035562  000240                        NOP                 ; (FOR PATCHING)
6116 035564  004737  003346      22$:      JSR     PC,MASCLR   ;RESET THE DMV
6117 035570  103001                        BCC     23$         ; RESET OK, PROCEED
6118 035572                                ERROR               ; RESET FAILED, REPORT IT
     035572  104460                                                                     TRAP    C$ERROR
6119 035574                          23$:
6120 035574                                ESCAPE  SUB         ; & GET THE HECK OUT OF HERE!!!
     035574  104410                                                                     TRAP    C$ESCAPE
     035576  000160                                                                     .WORD   L10067-.
6121 035600                          HLTST2:
6122
6123 035600  004537  005004                JSR     R5,MOVSW    ;RESTORE THE INTERRUPT VECTORS
6124 035604  002654                        BUFAREA             ; FROM THE BUFFER AREA
6125 035606  000000                        0
6126 035610  000200                        400/2               ; TO LOC'S 0 ==> 377
6127
6128 035612  010002                        MOV     R0,R2       ;R0 WILL BE CORRUPTED BY SUPERVISOR CALLS!
6129
6130 035614                                SETPRI  #0          ;RESTORE PRIORITY LEVEL TO 0
     035614  012700  000000                                                             MOV     #0,R0
     035620  104441                                                                     TRAP    C$SPRI
6131 035622  022701  071573                CMP     #071573,R1  ;CHECK RESULTS OF THE ADDITIONS
6132 035626  001412                        BEQ     5$          ;OK -->
6133 035630  010137  002256                MOV     R1,BDATA    ;WHAT!!!  SETUP & REPORT AN
6134 035634  012737  177777  002254        MOV     #-1,GDATA   ;  ADDITION ERROR
6135 035642                                GEDF    EM43D,ERR1  ;ADDITION DIDN'T WORK PROPERLY
                                                                ;       "DEVICE FATAL" ERROR # 84
     035642  104455                                                                     TRAP    C$ERDF
     035644  000124                                                                     .WORD   84
     035646  017242                                                                     .WORD   EM43D
     035650  006212                                                                     .WORD   ERR1
6136 035652  000240                        NOP                 ; (FOR PATCHING)
6137 035654  020227  177777          5$:      CMP     R2,#-1      ;THE LOADED ROUTINE SHOULD HAVE DONE THIS
6138 035660  001003                        BNE     8$          ;IT DIDN'T  - IT FAILED
```

TEST 13 -- HALT MODE VERIFICATION

```
6139 035662  023702  002412              CMP     TMP0,R2         ;THESE SHOULD BE EQUAL TOO
6140 035666  001412                      BEQ     10$             ;OK -->
6141 035670  010237  002256        8$:   MOV     R2,BDATA        ;WHAT!!!  SETUP & REPORT AN ERROR
6142 035674  012737  177777  002254    MOV     #-1,GDATA
6143 035702                            GEDF    EM43D,ERR1      ;LOADED ROUTINE FAILED
                                                               ;       "DEVICE FATAL" ERROR # 85
     035702  104455                                                      TRAP    C$ERDF
     035704  000125                                                      .WORD   85
     035706  017242                                                      .WORD   EM43D
     035710  006212                                                      .WORD   ERR1
6144 035712  000240                      NOP                     ; (FOR PATCHING)
6145 035714  020637  002454        10$:  CMP     SP,OLDSP        ;THESE SHOULD ALSO BE EQUAL
6146 035720  001412                      BEQ     15$             ;OK -->
6147 035722  010637  002256              MOV     SP,BDATA        ;WHAT!!!  SETUP & REPORT AN ERROR
6148 035726  013737  002454  002254      MOV     OLDSP,GDATA
6149 035734                            GEDF    EM43D,ERR1      ;LOADED ROUTINE FAILED
                                                               ;       "DEVICE FATAL" ERROR # 86
     035734  104455                                                      TRAP    C$ERDF
     035736  000126                                                      .WORD   86
     035740  017242                                                      .WORD   EM43D
     035742  006212                                                      .WORD   ERR1
6150 035744  000240                      NOP                     ; (FOR PATCHING)
6151 035746  004737  003346        15$:  JSR     PC,MASCLR       ;MAKE SURE THE DMV IS RESET
6152 035752  103001                      BCC     16$             ; RESET OK, CONTINUE
6153 035754                            ERROR                   ;RESET FAILED, REPORT ERROR
     035754  104460                                                      TRAP    C$ERROR
6154 035756                        16$:
6155 035756                            ENDSUB
     035756                                                    L10067:
     035756  104403                                                      TRAP    C$ESUB
6156 035760                            ENDTST
     035760                                                    L10065:
     035760  104401                                                      TRAP    C$ETST
```

```
6158                                    .SBTTL  SUBTEST 1'S M-CODE -- ASSIGNMENTS
6159
6160                                    ;*****************************************************************************
6161                                    ;       MICRO-CODE FOR SUBROUTINE # 1
6162                                    ;*****************************************************************************
6163
6164 035762                            MC1:
6165                                    ;       ASSEMBLED BY:  COMPAS MICROSYSTEMS MINMIC (V6A)
6166                                    ;                       (WITH CHANGES EDITED IN)
6167                                    ;
6168                                    ;LINE# LOC   CODE       LINE
6169                                    ;0002  0000             *=$0000
6170                                    ;0003  0000
6171                                    ;0004  0000             ;EQUATES FOR BIT DEFINITIONS
6172                                    ;0005  0000             BIT0    =@1
6173                                    ;0006  0000             BIT1    =@2
6174                                    ;0007  0000             BIT2    =@4
6175                                    ;0008  0000             BIT4    =@20
6176                                    ;0009  0000             BIT5    =@40
6177                                    ;0010  0000             BIT6    =@100
6178                                    ;0011  0000
6179                                    ;0012  0000
6180                                    ;0013  0000             ;ADDRESS EQUATES FOR CSR REGISTERS
6181                                    ;0014  0000             BSEL4   =$14
6182                                    ;0015  0000             BSEL5   =BSEL4+1
6183                                    ;0016  0000             BSEL7   =BSEL4+3
6184                                    ;0017  0000
6185                                    ;0018  0000
6186                                    ;0019  0000             ;NPR ADDRESS REGISTER EQUATES
6187                                    ;0020  0000             NPRAIL  = $003C     ;IN NPR ADRS LO REG
6188                                    ;0021  0000             NPRAIH  = NPRAIL+1  ;IN NPR ADRS HI REG
6189                                    ;0022  0000             NPRAIX  = NPRAIL+2  ;IN NPR EXTENDED ADRS REG
6190                                    ;0023  0000
6191                                    ;0024  0000
6192                                    ;0025  0000             ;NPR DATA REG EQUATES
6193                                    ;0026  0000             NPRDIL  = $A600     ;IN NPR DATA LO REG
6194                                    ;0027  0000             NPRDIH  = NPRDIL+1  ;IN NPR DATA HI REG
6195                                    ;0028  0000
6196                                    ;0029  0000
6197                                    ;0030  0000             ;NPR CONTROL REG EQUATES
6198                                    ;0031  0000             NPRCTL  = $A604 ;NPR CONTROL REGISTER
6199                                    ;0032  0000             NONPR   = BIT6  ;USED TO PREVENT AN NPR
6200                                    ;0033  0000             INOUT   = BIT5  ;SET TO 1 FOR INPUT, SET TO 0 FOR OUTPUT NPR
6201                                    ;0034  0000             HALT    = BIT4  ;SET DURING MOP MODE ONLY
6202                                    ;0035  0000             PWRUP   = BIT2  ;CLEARED BY BUS INIT TO INDICATE PWR UP
6203                                    ;0036  0000             SDCLOW  = BIT1  ;SET TO 1 TO RESET LSI-11 FOR MOP BOOT
6204                                    ;0037  0000             DISINI  = BIT0  ;SET TO 1 TO DISABLE BUS INIT TO 6502
6205                                    ;0038  0000
6206                                    ;0039  0000
6207                                    ;0040  0000             ;NPR REQUEST FUNCTIONS
6208                                    ;0041  0000             NPRRED  = PWRUP ;IN/OUT BIT = 0 FOR READ TO DMV-11
6209                                    ;0042  0000
6210                                    ;0043  0000
6211                                    ;0044  0000             ;MISCELLANEOUS EQUATES
6212                                    ;0045  0000             STARAM  = $003F ;STARTING ADRS OF GEN'L PURPOSE RAM TO TEST
6213                                    ;0046  0000
6214                                    ;0047  0000
```

```
6215                                    .SBTTL  SUBTEST 1'S M-CODE -- ROUTINE
6216                                    ;0048  0000             **STARAM        ;START OF MICROCODE IN RAM
6217
6218                                    ;LINE# LOC   CODE       LINE
6219
6220 035762    251     000             .BYTE  251,00
6221                                    ;0050  003F  A9 00              LDA     #0      ;CLEAR BSEL7
6222 035764    205     027             .BYTE  205,27
6223                                    ;0051  0041  85 17              STA     BSEL7
6224                                    ;0052  0043
6225 035766    245     027             .BYTE  245,27
6226                                    ;0053  0043  A5 17      WAIT1   LDA     BSEL7   ;WAIT FOR IT TO GO <> 0
6227 035770    360     374             .BYTE  360,374
6228                                    ;0054  0045  F0 FC              BEQ     WAIT1
6229                                    ;0055  0047
6230                                    ;0056  0047             ; WE SHOULD NOW BE IN SYNC WITH THE 11 PROCESSOR
6231                                    ;0057  0047
6232 035772    245     024             .BYTE  245,24
6233                                    ;0058  0047  A5 14              LDA     BSEL4   ;GET & SAVE THE ADDRESS
6234 035774    205     074             .BYTE  205,74
6235                                    ;0059  0049  85 3C              STA     NPRAIL  ;OF "TMP0" AND USE IT TO
6236 035776    245     025             .BYTE  245,25
6237                                    ;0060  004B  A5 15              LDA     BSEL5   ;SETUP FOR AN NPR-IN
6238 036000    205     075             .BYTE  205,75
6239                                    ;0061  004D  85 3D              STA     NPRAIH  ;OPERATION LATER
6240                                    ;0062  004F
6241 036002    251     124             .BYTE  251,124
6242                                    ;0063  004F  A9 54              LDA     #NONPR!HALT!PWRUP
6243 036004    215     004     246     .BYTE  215,4,246
6244                                    ;0064  0051  8D 04 A6           STA     NPRCTL  ;HALT THE 11 CPU
6245                                    ;0064  0054
6246                                    ;0064  0054             ; DELAY TO ALLOW "HALT" TO TAKE EFFECT (ABOUT
6247                                    ;0064  0054             ; 100 MICROSECONDS).
6248 036007    240     041             .BYTE  240,41
6249                                    ;0064  0054  A0 21              LDY     #$21    ;INITIAL VALUE OF COUNTER
6250 036011    210                     .BYTE  210
6251                                    ;0064  0056  88         DELAY   DEY             ;(33. FOR .6 US CYCLE)
6252 036012    320     375             .BYTE  320,375
6253                                    ;0064  0057  D0 FD              BNE     DELAY
6254                                    ;0065  0059
6255                                    ;0066  0059             ; WE NOW HAVE TO READ THE 11 CPU'S LOCATION WHO'S
6256                                    ;0067  0059             ; ADDRESS WE PREVIOUSLY READ FROM SEL6
6257                                    ;0068  0059
6258 036014    251     000             .BYTE  251,00
6259                                    ;0069  0059  A9 00              LDA     #0      ;CLEAR THE EXTENDED-ADDRESS-IN
6260 036016    205     076             .BYTE  205,76
6261                                    ;0070  005B  85 3E              STA     NPRAIX
6262 036020    251     024             .BYTE  251,24
6263                                    ;0071  005D  A9 14              LDA     #NPRRED!HALT
6264 036022    215     004     246     .BYTE  215,4,246
6265                                    ;007?  005F  8D 04 A6           STA     NPRCTL  ;READ ONE WORD FROM THE 11 CPU
6266                                    ;007?  0062
6267 036025    054     004     246     .BYTE  54,4,246
6268                                    ;0074  0062  2C 04 A6   NPRWAT  BIT     NPRCTL  ;WAIT FOR IT TO "ALMOST" COMPLETE
6269 036030    160     373             .BYTE  160,373
6270                                    ;0075  0065  70 FB              BVS     NPRWAT
6271 036032    352                     .BYTE  352
```

```
6272                                    ;0075  0067  EA              NOP             ;SHOULD COMPLETE HERE
6273                                    ;0076  0068
6274 036033    255     000     246      .BYTE  255,0,246
6275                                    ;0077  0068  AD 00 A6        LDA     NPRDIL  ;MOVE THE WORD JUST READ INTO
6276 036036    205     024              .BYTE  205,24
6277                                    ;0078  006B  85 14           STA     BSEL4   ; SEL4
6278 036040    255     001     246      .BYTE  255,1,246
6279                                    ;0079  006D  AD 01 A6        LDA     NPRDIH
6280 036043    205     025              .BYTE  205,25
6281                                    ;0080  0070  85 15           STA     BSEL5
6282                                    ;0088  0072
6283                                    ;0089  0072             ; DROP "HALT" AND SET "DCOK H LC" & "DISABL INIT"
6284                                    ;0090  0072
6285 036045    251     125              .BYTE  251,125
6286                                    ;0091  0072  A9 55           LDA     #NONPR!PWRUP!DISINI!HALT
6287 036047    215     004     246      .BYTE  215,4,246
6288                                    ;0092  0074  8D 04 A6        STA     NPCRTL
6289 036052    251     107              .BYTE  251,107
6290                                    ;0093  0077  A9 47           LDA     #NONPR!PWRUP!SDCLOW!DISINI
6291 036054    215     004     246      .BYTE  215,4,246
6292                                    ;0094  0079  8D 04 A6        STA     NPRCTL
6293                                    ;0095  007C
6294                                    ;0096  007C             ; NOW LET THE 11 CPU GO THROUGH THE POWER-UP SEQUENCE
6295                                    ;0097  007C
6296 036057    251     105              .BYTE  251,105
6297                                    ;0098  007C  A9 45           LDA     #NONPR!PWRUP!DISINI
6298 036061    215     004     246      .BYTE  215,4,246
6299                                    ;0099  007E  8D 04 A6        STA     NPRCTL
6300                                    ;0100  0081
6301                                    ;0101  0081             ; WHEN BSEL7 IS CLEARED, CLEAR "DISABL INIT"
6302                                    ;0102  0081
6303 036064    245     027              .BYTE  245,27
6304                                    ;0103  0081  A5 17      WAIT2   LDA     BSEL7
6305 036066    320     374              .BYTE  320,374
6306                                    ;0104  0083  D0 FC           BNE     WAIT2
6307                                    ;0105  0085
6308 036070    251     104              .BYTE  251,104
6309                                    ;0106  0085  A9 44           LDA     #NONPR!PWRUP
6310 036072    215     004     246      .BYTE  215,4,246
6311                                    ;0107  0087  8D 04 A6        STA     NPRCTL
6312                                    ;0108  008A
6313                                    ;0109  008A             ; USE A STANDARD SUBROUTINE RETURN TO GET BACK INTO
6314                                    ;0110  008A             ; THE MAINTENANCE LOOP
6315                                    ;0111  008A
6316 036075    140                      .BYTE  140
6317                                    ;0112  008A  60              RTS
6318                                    ;0113  008B
6319
6320
6321                                    ;ERRORS = 0000
6322
6323                                    .SBTTL  SUBTEST 1'S M-CODE -- SYMBOL   TABLE
6324
6325                                    ; BIT0    0001    BIT1    0002    BIT2    0004    BIT4    0010
6326                                    ; BIT5    0020    BIT6    0040    BSEL4   0014    BSEL5   0015
6327                                    ; BSEL7   0017    DELAY   006E    DISINI  0001    HALT    0010
6328                                    ; INOUT   0020    NONPR   0040    NPRAIH  003D    NPRAIL  003C
```

```
6329                    ; NPRAIX   003E    NPRCTL   A604    NPRDIH   A601    NPRDIL   A600
6330                    ; NPRRED   0004    NPRWAT   005D    PWRUP    0004    SDCLOW   0002
6331                    ; STARAM   003F    WAIT1    0043    WAIT2    0078
6332                    ;END OF ASSEMBLY(V6A)
6333                    ;SYMBOLS LEFT = 1473 OUT OF 1500
6334
6335                    .SBTTL  SUBTEST 1'S M-CODE -- CROSS REFERENCE TABLE (CREF V01-05 )
6336
6337
6338                    ;BIT0              5#      37
6339                    ;BIT1      6#      36
6340                    ;BIT2      7#      35
6341                    ;BIT4      8#      34
6342                    ;BIT5      9#      33
6343                    ;BIT6     10#      32
6344                    ;BSEL4    14#      58      78
6345                    ;BSEL5    15#      60      80
6346                    ;BSEL7    16#      15      16      51      53      100
6347                    ;DELAY    86#      87
6348                    ;DISINI   37#      91      96
6349                    ;HALT              34#     63      71
6350                    ;INOUT    33#
6351                    ;NONPR    32#      63      91      96      103
6352                    ;NPRAIH   21#      61
6353                    ;NPRAIL   20#      21      22      59
6354                    ;NPRAIX   22#      70
6355                    ;NPRCTL   31#      64      72      74      92      104
6356                    ;NPRDIH   27#      79
6357                    ;NPRDIL   26#      27      77
6358                    ;NPRRED   41#      71
6359                    ;NPRWAT   74#      75
6360                    ;PWRUP    35#      41      63      91      96      103
6361                    ;SDCLOW   36#      91
6362                    ;STARAM   45#      48
6363                    ;WAIT1    53#      54
6364                    ;WAIT2   100#     101
```

```
                                .EVEN
                                .SBTTL  SUBTEST 2'S M CODE - ASSIGNMENTS

                                ;*****************************************************************************
                                ;       MICRO-CODE FOR SUBROUTINE 0 2
                                ;*****************************************************************************
                        000010  .RADIX  8.
6360'6                  MC2:
                                ;LINE# LOC   CODE        LINE
                                ;0002  0000              ;;$0000
                                ;0003  0000
                                ;0004  0000              ;EQUATES FOR BIT DEFINITIONS
                                ;0005  0000              BIT0    =@1
                                ;0006  0000              BIT1    =@2
                                ;0007  0000              BIT2    =@4
                                ;0008  0000              BIT4    =@20
                                ;0009  0000              BIT6    =@100
                                ;0010  0000
                                ;0011  0000
                                ;0012  0000              ;ADDRESS EQUATES FOR CSR REGISTERS
                                ;0013  0000              BSEL7   =$17
                                ;0014  0000
                                ;0015  0000
                                ;0016  0000              ;NPR ADDRESS REGISTER EQUATES
                                ;0017  0000              NPRA0L  = $0038 ;OUT NPR ADRS LO REG
                                ;0018  0000              NPRA0H  = NPRA0L+1 ;OUT NPR ADRS HI REG
                                ;0019  0000              NPRA0X  = NPRA0L+2 ;OUT NPR EXTENDED ADRS REG
                                ;0020  0000
                                ;0021  0000
                                ;0022  0000              ;NPR DATA REG EQUATES
                                ;0023  0000              NPRD0L  = $A600 ;OUT NPR DATA LO REG
                                ;0024  0000              NPRD0H  = NPRD0L+1 ;OUT NPR DATA HI REG
                                ;0025  0000
                                ;0026  0000
                                ;0027  0000              ;NPR CONTROL REG EQUATES
                                ;0028  0000              NPRCTL  = $A604 ;NPR CONTROL REGISTER
                                ;0029  0000              NONPR   = BIT6  ;USED TO PREVENT AN NPR
                                ;0030  0000              HALT    = BIT4  ;SET DURING MOP MODE ONLY
                                ;0031  0000              PWRUP   = BIT3  ;CLEARED BY BUS INIT
                                ;0032  0000              SDCLOW  = BIT1  ;SET TO 1 TO RESET LSI-11 FOR MOP BOOT
                                ;0033  0000              DISINI  = BIT0  ;SET TO 1 TO DISABLE BUS INIT TO KMC
                                ;0034  0000
                                ;0035  0000
                                ;0036  0000              ;MISCELLANEOUS EQUATES
                                ;0037  0000              STARAM  = 1003F ;STARTING ADRS OF GEN'L PURPOSE RAM
                                ;0038  0000
```

SUBTEST .'S M CODE ROUTINE

```
6413                                                    .SBTTL  SUBTEST .'S M CODE     ROUTINE
6414                                                    ;0039  0000
6415                                                    ;0040  0000                 A**STARAM       ;START OF MICROCODE IN RAM
6416 036076     251     000                             .BYTE  251,000
6417                                                    ;0042  003F  A9 00                  LDA     #0      ;CLEAR BSEL7
6418 036100     205     027                             .BYTE  205,027
6419                                                    ;0043  0041  85 17                  STA     BSEL7
6420                                                    ;0044  0043
6421 036102     245     027                             .BYTE  245,027
6422                                                    ;0045  0043  A5 17          WAIT1   LDA     BSEL7   ;WAIT FOR IT TO GO <- 0
6423 036104     360     374                             .BYTE  360,374
6424                                                    ;0046  0045  F0 FC                  BEQ     WAIT1
6425                                                    ;0047  0047
6426                                                    ;0048  0047                 ; WE SHOULD NOW BE IN SYNC WITH THE 11 PROCESSOR
6427                                                    ;0049  0047
6428                                                    ;0050  0047                 ; THE NEXT SEQUENCE WILL SEND THE 11 CPU THROUGH
6429                                                    ;0051  0047                 ; A POWER-UP SEQUENCE AND RESET EVERYTHING ELSE ON THE
6430                                                    ;0052  0047                 ; Q-BUS.  WE HAVE PREVENTED OURSELVES FROM BEING RESET
6431                                                    ;0053  0047                 ; BY SETTING "DISABL INIT".
6432                                                    ;0054  0047
6433 036106     251     125                             .BYTE  251,125
6434                                                    ;0054  0047  A9 55                  LDA     #NONPR!HALT!PWRUP!DISINI
6435 036110     215     004     246                     .BYTE  215,004,246
6436                                                    ;0054  0049  8D 04 A6               STA     NPRCTL
6437                                                    ;0054  004C
6438                                                    ;0054  004C                 ; DELAY TO ALLOW "HALT" TO TAKE EFFECT (ABOUT
6439                                                    ;0054  004C                 ; 100 MICROSECONDS).
6440 036113     240     041                             .BYTE  240,41
6441                                                    ;0054  004C  A0 21                  LDY     #21     ;INITIAL VALUE OF COUNTER
6442 036115     210                                     .BYTE  210
6443                                                    ;0054  004E  88             DELAY   DEY             ;(33. FOR .6 US CYCLE)
6444 036116     320     375                             .BYTE  320,375
6445                                                    ;0054  004F  D0 FD                  BNE     DELAY
6446 036120     251     127                             .BYTE  251,127
6447                                                    ;0055  0051  A9 57                  LDA     #NONPR!HALT!PWRUP!SDCLOW!DISINI
6448 036122     215     004     246                     .BYTE  215,004,246
6449                                                    ;0056  0053  8D 04 A6               STA     NPRCTL  ;HANG THE 11 CPU ETC.
6450 036125     251     165                             .BYTE  251,165
6451                                                    ;0057  0056  A9 75                  LDA     #NONPR!HALT!PWRUP!DISINI!NPROUT
6452 036127     215     004     246                     .BYTE  215,004,246
6453                                                    ;0058  0058  8D 04 A6               STA     NPRCTL  ;NOW LET IT  POWER UP
6454                                                    ;0059  0058
6455 036132     251     000                             .BYTE  251,000
6456                                                    ;0060  0058  A9 00                  LDA     #0      ;SETUP NPR ADDR OUT HIGH FOR
6457 036134     205     071                             .BYTE  205,071
6458                                                    ;0061  0050  85 [illegible]         STA     NPRADH  ;  ALL NPR'S
6459 036136     205     072                             .BYTE  205,072
6460                                                    ;0062  005F  85 5A                  STA     NPRADX  ;THE EXTENDED BITS TOO
6461 036140     252                                     .BYTE  252
6462                                                    ;0063  0061  AA                     TAX             ;INITALIZE DATA TABLE INDEX
6463                                                    ;0064  0062
6464                                                    ;0065  0062                 ; WE ARE NOW SETUP TO MOVE THE  MOV  INSTRUCTION INTO
6465                                                    ;0066  0062                 ; LOCATION 0 OF THE 11'S MEMORY
6466                                                    ;0067  0062
6467 036141     251     024                             .BYTE  251,024
6468                                                    ;0068  0062  A9 14                  LDA     #024    ;POINT TO 11'S POWER UP VEC.
6469 036143     205     070                             .BYTE  205,070
```

```
SUBTEST #'S M CODE     ROUTINE

6470
6471 036145   040   305   000          ;0069  0064  85 38          STA     NPRADR
                                       .BYTE  040,305,000
6472                                   ;0070  0066  20 C5 00       JSR     NPR     ;MOVE TO 24
6473 036150   040   305   000          .BYTE  040,305,000
6474                                   ;0071  0069  20 C5 00       JSR     NPR     ;MOVE TO 26
6475                                   ;0072  006C
6476 036153   251   000                .BYTE  251,000
6477                                   ;0073  006C  A9 00          LDA     #0
6478 036155   205   070                .BYTE  205,070
6479                                   ;0074  006E  85 38          STA     NPRADR  ;POINT TO THE 11'S LOC. 0
6480 036157   040   305   000          .BYTE  040,305,000
6481                                   ;0075  0070  20 C5 00       JSR     NPR     ;MOVE TO LOC 0
6482 036162   040   305   000          .BYTE  040,305,000
6483                                   ;0076  0073  20 C5 00       JSR     NPR     ;MOVE TO LOC 2
6484 036165   040   305   000          .BYTE  040,305,000
6485                                   ;0077  0076  20 C5 00       JSR     NPR     ;MOVE TO 4
6486                                   ;0078  0079
6487                                   ;0079  0079           ; NOW THAT WE'VE TAKEN CONTROL OF THE 11'S POWER
6488                                   ;0080  0079           ; UP VECTOR, FORCE HIM THROUGH IT.  BUT DON'T LET
6489                                   ;0081  0079           ; HIS BINIT RESET US (BY KEEPING "DISABL INIT"
6490                                   ;0082  0079           ; SET)
6491                                   ;0083  0079
6492 036170   251   107                .BYTE  251,107
6493                                   ;0084  0079  A9 47          LDA     #NONPR!PWRUP!SDCLOW!DISINI
6494 036172   215   004   246          .BYTE  215,004,246
6495                                   ;0085  0078  8D 04 A6       STA     NPRCTL
6496 036175   251   145                .BYTE  251,145
6497                                   ;0086  007E  A9 65          LDA     #NONPR!PWRUP!DISINI!NPROUT
6498 036177   215   004   246          .BYTE  215,004,246
6499                                   ;0087  0080  8D 04 A6       STA     NPRCTL
6500 036202   352                     .BYTE  352
6501                                   ;0087  0083  EA             NOP
6502 036203   352                     .BYTE  352
6503                                   ;0087  0084  EA             NOP
6504                                   ;0088  0085
6505                                   ;0089  0085           ; NOW WE CAN NPR OUT THE REST OF THE DATA WE HAVE
6506                                   ;0090  0085
6507 036204   040   305   000          .BYTE  040,305,000
6508                                   ;0091  0085  20 C5 00       JSR     NPR     ;MOVE TO LOC 6
6509 036207   240   166                .BYTE  240,166
6510                                   ;0092  0088  A0 76          LDY     #@166   ;THIS IS HOW MANY WE'LL DO
6511 036211   040   305   000          .BYTE  040,305,000
6512                                   ;0093  008A  20 C5 00  FILOOP  JSR     NPR     ;WRITE 1 WORD
6513 036214   312                     .BYTE  312
6514                                   ;0094  008D  CA             DEX             ;BACK UP THE DATA POINTER
6515 036215   312                     .BYTE  312
6516                                   ;0095  008E  CA             DEX             ;    WE WANT THE SAME WORD
6517 036216   210                     .BYTE  210
6518                                   ;0096  008F  88             DEY             ;IF NOT DONE
6519 036217   320   370                .BYTE  320,370
6520                                   ;0097  0090  D0 F8          BNE     FILOOP  ;                DO IT AGAIN
6521 036221   240   006                .BYTE  240,006
6522                                   ;0098  0092  A0 06          LDY     #6      ;ELSE, SETUP TO DO 6 MORE
6523 036223   350                     .BYTE  350
6524                                   ;0099  0094  E8             INX             ;       WORDS
6525 036224   350                     .BYTE  350
6526                                   ;0100  0095  E8             INX             ;POINT TO NEXT WORD
```

```
6527 036225  040  305  000   .BYTE  040,305,000
6528                       ;0101  0096  20 C5 00   ENDLOP  JSR     NPR       ;MOVE THE LAST 6 WORDS
6529 036230  210                .BYTE  210
6530                       ;0102  0099  88                 DEY               ;IF NOT DONE,
6531 036231  320  372            .BYTE  320,372
6532                       ;0103  009A  D0 FA              BNE     ENDLOP    ;                 DO IT AGAIN
6533                       ;0104  009C                                       ;ELSE, THE SIMULATED "DOWN
6534                       ;0105  009C                                       ;  LINE LOAD" IS COMPLETE
6535                       ;0106  009C
6536 036233  251  004            .BYTE  251,004
6537                       ;0107  009C  A9 04              LDA     #4        ; AND THE 11'S LOC. 4
6538 036235  205  070            .BYTE  205,070
6539                       ;0108  009E  85 38              STA     NPRAOI
6540 036237  040  305  000   .BYTE  040,305,000
6541                       ;0109  00A0  20 C5 00           JSR     NPR       ;OVER-WRITE THE "BR  ."
6542                       ;0110  00A3                                       ;INSTRUCTION TO LET THE JUST
6543                       ;0111  00A3                                       ;LOADED ROUTINE BE EXECUTED
6544                       ;0112  00A3
6545 036242  251  104            .BYTE  251,104
6546                       ;0113  00A3  A9 44              LDA     #NONPR!PWRUP
6547 036244  215  004  246   .BYTE  215,004,246
6548                       ;0114  00A5  8D 04 A6           STA     NPRCTL    ;LET BINIT RESET US AGAIN
6549                       ;0115  00A8
6550 036247  140                .BYTE  140
6551                       ;0116  00A8  60                 RTS               ;RETURN TO MAINTENANCE LOOP
6552                       ;0117  00A9
6553                       ;0118  00A9
6554                       ;0119  00A9          ;*****************************************************
6555                       ;0120  00A9          ;
6556                       ;0121  00A9          ;       DATABL -- DATA TABLE CONTAINING THE DATA THAT
6557                       ;0122  00A9          ;                 IS TO BE NPR'D INTO THE 11'S MEMORY
6558                       ;0123  00A9          ;
6559                       ;0124  00A9          ;*****************************************************
6560                       ;0125  00A9
6561                       ;0126  00A9
6562                       ;0127  00A9          DATABL
6563 036250  000000           .WORD  000000
6564                       ;0128  00A9  00 00              .DBYTE  0,@340    ;LOC'S 24 & 26
6565 036252  000340           .WORD  000340
6566                       ;0128  00AB  00 E0
6567 036254  012700  177777   MOV     #-1,R0
6568                       ;0129  00AD  15 C0              .DBYTE  @012700, 1,@777 ;LOC'S 0   - 4
6569                       ;0129  00AF  FF FF
6570 036260  000777           BR      .
6571                       ;0129  00B1  01 ??
6572 036262  005001           CLR     R1
6573                       ;0130  00B3  0A 01              .DBYTE  @005001   ;LOC. 6
6574 036264  062701           .WORD  062701
6575                       ;0131  00B5  65 C1              .DBYTE  @062701   ;LOC'S 10  - > 362
6576 036266  010037  002412   MOV     R0,@#TMP0
6577                       ;0132  00B7  10 1F              .DBYTE  @010037   ;LOC 364   "MOV
6578                       ;0133  00B9  00 00              .DBYTE  0         ;LOC 366
6579 036272  013706  002454   MOV     @#OLDSP,SP
6580                       ;0134  00BB  17 C6              .DBYTE  @013706   ;LOC 370   "MOV
6581                       ;0135  00BD  00 00              .DBYTE  0         ;LOC 372
6582 036276  000137  035600   JMP     @#HLTST2
6583                       ;0136  00BF  00 5F              .DBYTE  @000137   ;LOC 374   "JMP"
```

```
6584
6585 036302  000240                ;0137  00C1  00 00              .DBYTE  0        ;LOC 376
                                    NOP
6586                                ;0138  00C3  00 A0              .DBYTE  @000240 ;"NOP" FOR LOC 4
6587                                ;0139  00C5
6588                                ;0140  00C5                ;  THE THREE WORDS FOR LOCATIONS 366, 372, & 376 ARE
6589                                ;0141  00C5                ;  ASSEMBLED IN WHEN THIS CODE IS INCLUDED INTO THE
6590                                ;0142  00C5                ;  DIAGNOSTIC
6591                                ;0143  00C5
6592                                ;0144  00C5                ;*****************************************************
6593                                ;0145  00C5                ;
6594                                ;0146  00C5                ;       "NPR" SUBROUTINE:
6595                                ;0147  00C5                ;
6596                                ;0148  00C5                ; 1 TAKE THE DATA FROM THE DATA TABLE AS INDEXED BY
6597                                ;0149  00C5                ;       "X" AND PUT IT INTO THE NPR DATA OUT REGISTERS
6598                                ;0150  00C5                ;
6599                                ;0151  00C5                ; 2 GET THE CURRENT SETTING OF THE NPR CONTROL REG.
6600                                ;0152  00C5                ;       AND WRITE IT BACK TO CAUSE A WORD NPR-OUT
6601                                ;0153  00C5                ;
6602                                ;0154  00C5                ; 3 INCREMENT THE NPR-OUT-ADDRESS-LOW REGISTER
6603                                ;0155  00C5                ;
6604                                ;0156  00C5                ; 4 WAIT FOR "GOBUSY" TO GO LOW
6605                                ;0157  00C5                ;
6606                                ;0158  00C5                ; 5 RETURN TO CALLER
6607                                ;0159  00C5                ;
6608                                ;0160  00C5                ;*****************************************************
6609                                ;0161  00C5
6610 036304  265  251                .BYTE  265,251
6611                                ;0162  00C5  B5 A9      NPR     LDA     DATABL,X ;LOAD THE DATA-OUT REG'S
6612 036306  215  000  246           .BYTE  215,000,246
6613                                ;0163  00C7  8D 00 A6           STA     NPRDOL
6614 036311  265  252                .BYTE  265,252
6615                                ;0164  00CA  B5 AA              LDA     DATABL+1,X
6616 036313  215  001  246           .BYTE  215,001,246
6617                                ;0165  00CC  8D 01 A6           STA     NPRDOH
6618                                ;0166  00CF
6619 036316  255  004  246           .BYTE  255,004,246
6620                                ;0167  00CF  AD 04 A6           LDA     NPRCTL
6621 036321  215  004  246           .BYTE  215,004,246
6622                                ;0168  00D2  8D 04 A6           STA     NPRCTL  ;KICK OFF A WORD NPR OUT
6623                                ;0169  00D5
6624 036324  350                     .BYTE  350
6625                                ;0170  00D5  E8                 INX             ;POINT TO THE NEXT DATA
6626 036325  350                     .BYTE  350
6627                                ;0171  00D6  E8                 INX             ;  WORD
6628                                ;0172  00D7
6629                                ;0173  00D7
6630                                ;0174  00D7             NPRWAT
6631 036326  054  004  246           .BYTE  054,004,246
6632                                ;0175  00D7  2C 04 A6           BIT     NPRCTL  ;WAIT FOR THE NPR TO
6633 036331  160  373                .BYTE  160,373
6634                                ;0176  00DA  70 FB              BVS     NPRWAT  ; COMPLETE
6635 036333  352                     .BYTE  352
6636                                ;0177  00DC  EA                 NOP
6637 036334  346  070                .BYTE  346,070
6638                                ;0178  00DD  E6 38              INC     NPRAOL  ;POINT TO THE NEXT WORD
6639 036336  346  070                .BYTE  346,070
6640                                ;0179  00DF  E6 38              INC     NPRAOL  ;OF THE 11'S MEMORY
```

```
6641
6642 036340      140                    ;0180  00E1
                                         .BYTE  140
6643                                     ;0181  00E1  60                RTS              ;RETURN TO CALLER
6644                                     ;0182  00E2
6645                                     ;
6646                                     ;
6647                                     ;ERRORS = 0000
6648                                     ;
6649                                     .SBTTL  SUBTEST 2'S M-CODE -- SYMBOL  TABLE
6650                                     ;
6651                                     ; BIT0     0001    BIT1     0002    BIT2     0004    BIT4     0010
6652                                     ; BIT6     0040    BSEL7    0017    DATABL   009D    DISINI   0001
6653                                     ; ENDLOP   008A    FILOOP   007E    HALT     0010    NONPR    0040
6654                                     ; NPR      0089    NPRAOH   0039    NPRAOL   0038    NPRAOX   003A
6655                                     ; NPRCTL   A604    NPRDOH   A601    NPRDOL   A600    NPRWAT   00CB
6656                                     ; PWRUP    0004    SDCLOW   0002    STARAM   003F    WAIT1    0043
6657                                     ;
6658                                     ;END OF ASSEMBLY(V6A)
6659                                     ;SYMBOLS LEFT = 1476 OUT OF 1500
6660                                     ;
6661                                     ;COMPAS MICROSYSTEMS MINMIC CROSS ASSEMBLER PAGE C 1
6662                                     .SBTTL  SUBTEST 2'S M-CODE -- CROSS REFERENCE TABLE (CREF V01.05 )
6663                                     ;
6664                                     ;
6665                                     ;BIT0     50     33
6666                                     ;BIT1     60     32
6667                                     ;BIT2     70     31
6668                                     ;BIT4     80     30
6669                                     ;BIT6     90     29
6670                                     ;BSEL7   130     43     45
6671                                     ;DATABL  1270    162    164
6672                                     ;DISINI   330     55     57     84     86
6673                                     ;ENDLOP  1010    103
6674                                     ;FILOOP   930     97
6675                                     ;HALT     300     55     57
6676                                     ;NONPR    290     55     57     84     86    113
6677                                     ;NPR       70     71     75     76     77     91     93
6678                                     ;        101    109   1620
6679                                     ;NPRAOH   180     61
6680                                     ;NPRAOL   170     18     19     69     74    108    178
6681                                     ;        179
6682                                     ;NPRAOX   190     62
6683                                     ;NPRCTL   280     56     58     85     87    114    167
6684                                     ;        168    175
6685                                     ;NPRDOH   240    165
6686                                     ;NPRDOL   230     24    163
6687                                     ;NPRWAT  1740    176
6688                                     ;PWRUP    310     55     57     84     86    113
6689                                     ;SDCLOW   320     55     84
6690                                     ;STARAM   370     40
6691                                     ;WAIT1    450     46
6692 036341                             MC2END:
6693                                     .EVEN
```

```
                                .SBTTL HARDWARE PARAMETER CODING SECTION


                                ;////////////////////////////////////////////////////////////////////////////
                                ;/ THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ;/ WITH THE OPERATOR.
                                ;////////////////////////////////////////////////////////////////////////////

036342                                  BGNHRD
036342  000025                                                                  .WORD L10070-L$HARD/2
036344                                                                  L$HARD::
                                
036344                                  GPRMA   ADDRES,0,0,160020,177776,YES
036344  000031                                                                  .WORD   T$CODE
036346  036416                                                                  .WORD   ADDRES
036350  160020                                                                  .WORD   T$LOLIM
036352  177776                                                                  .WORD   T$HILIM
036354                                  GPRMA   VECTOR,2,0,0,674,YES
036354  001031                                                                  .WORD   T$CODE
036356  036444                                                                  .WORD   VECTOR
036360  000000                                                                  .WORD   T$LOLIM
036362  000674                                                                  .WORD   T$HILIM
036364                                  GPRMD   PRIRTY,4,0,7000,0,7,YES
036364  002032                                                                  .WORD   T$CODE
036366  036475                                                                  .WORD   PRIRTY
036370  007000                                                                  .WORD   7000
036372  000000                                                                  .WORD   T$LOLIM
036374  000007                                                                  .WORD   T$HILIM
                                ;       GPRML   PU24.M,16,100,YES                       ;JB REV A-0
036376                                  GPRMD   BDTY.M,12,0,7,0,2,YES
036376  005032                                                                  .WORD   T$CODE
036400  036526                                                                  .WORD   BDTY.M
036402  000007                                                                  .WORD   7
036404  000000                                                                  .WORD   T$LOLIM
036406  000002                                                                  .WORD   T$HILIM
036410                                  GPRML   XMFG.M,16,200,YES
036410  007130                                                                  .WORD   T$CODE
036412  036673                                                                  .WORD   XMFG.M
036414  000200                                                                  .WORD   200

036416                                  ENDHRD
                                
                                                                                .EVEN
036416                                                                  L10070:

                                        .NLIST  BEX
036416  104  105  126  ADDRES: .ASCIZ  /DEVICE CSR ADDRESS : /
036444  104  105  126  VECTOR: .ASCIZ  /DEVICE VECTOR ADDRESS : /
036475  104  105  126  PRIRTY: .ASCIZ  /DEVICE PRIORITY LEVEL : /
036526  102  117  101  BDTY.M: .ASCIZ  /BOARD TYPE (0-M8064, 1-M8053 V.35, 2-M8053-EIA) : /
036611  111  123  040  PU24.M: .ASCIZ  /IS THE PROCESSOR STRAPPED TO MODE 0 ON POWER-UP? /
036673  111  123  040  XMFG.M: .ASCIZ  /IS THIS A MANUFACTURING TEST STAND?/
                                .LIST   BEX
```

HARDWARE PARAMETER CODING SECTION

[illegible] .EVEN

```
                                .SBTTL SOFTWARE PARAMETER CODING SECTION



                                ;////////////////////////////////////////////////////////////////////////
                                ;/  THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ;/  THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ;/  MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ;/  INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ;/  MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ;/  WITH THE OPERATOR.
                                ;////////////////////////////////////////////////////////////////////////

036740                                  BGNSFT
036740  000000                                                          .WORD L10071-L$SOFT/2
036742                                                          L$SOFT::
036742                                  ENDSFT
                                                                .EVEN
036742                                                          L10071:
```

PATCH AREA FOR DEBUG

```
6745                                    .SBTTL PATCH AREA FOR DEBUG
6746 036742                             PATCH:
6747            037042                          .=.+100
6748 037042     000240                          NOP
6749 037044     000240                          NOP
6750 037046     000240                          NOP
6751
6752                                    ;*****************************************
6753
6754                                    .SBTTL "ENDMOD" & "LASTAD"
6755 037050                                     ENDMOD
6756 037050                                     LASTAD

     037050     000000                                                  .EVEN
     037052     000000                                                  .WORD   0
     037054                                                             .WORD   0
                                        L$LAST::
6757            000001                  .END
```

SYMBOL TABLE

```
ACR     = 120013          BSLT3 = 000023          C$GETB= 000026          EM34    016526          ERR10$  012732
ADDRES  036416            BSLT4 = 000024          C$GETW= 000027          EM34B   016540          ERR10.  012752
ADR     = 000020 G        BSLT5 = 000025          C$GMAN= 000043          EM35    016552          ERR11   007314 G
AD.HIT  023070            BSLT6 = 000026          C$GPHR= 000042          EM35B   016573          ERR12   007456 G
AD.OK   023064            BSLT7 = 000027          C$GPLO= 000030          EM4     016135          ERR13   010240 G
ASSEMB= 000010            BSR0    002212          C$GPRI= 000040          EM40    016614          ERR14   011002 G
BDATA   002256            BSR1    002214          C$INIT= 000011          EM41    016671          ERR3    006220 G
BDTY.M  036526            BSR10   002232          C$INLP= 000020          EM42A   016746          ERR4    006232 G
BIT0    = 000001 G        BSR11   002234          C$MANI= 000050          EM42B   017020          ERR4$   012206
BIT00   = 000001 G        BSR12   002236          C$MEM = 000031          EM42C   017074          ERR51   011176 G
BIT01   = 000002 G        BSR13   002240          C$MSG = 000023          EM43A   017153          ERR52   011270 G
BIT02   = 000004 G        BSR14   002242          C$OPEN= 000034          EM43B   017177          ERR60   011720 G
BIT03   = 000010 G        BSR15   002244          C$PNTB= 000014          EM43C   017215          ERR8    006532 G
BIT04   = 000020 G        BSR16   002246          C$PNTF= 000017          EM43D   017242          ERR9    007300 G
BIT05   = 000040 G        BSR17   002250          C$PNTS= 000016          EM50A   017271          ERR9$   012574
BIT06   = 000100 G        BSR2    002216          C$PNTX= 000015          EM50B   017337          ERR9.   012614
BIT07   = 000200 G        BSR3    002220          C$QIO = 000377          EM50C   017405          EVL     = 000004 G
BIT08   = 000400 G        BSR4    002222          C$RDBU= 000007          EM50D   017453          EXECUT= 000005
BIT09   = 001000 G        BSR5    002224          C$REFG= 000047          EM50E   017507          E$END   = 002100
BIT1    = 000002 G        BSR6    002226          C$RESE= 000033          EM50F   017543          E$LOAD= 000035
BIT10   = 002000 G        BSR7    002230          C$REVI= 000003          EM50G   017610          FMT02A  013127
BIT11   = 004000 G        BT1     = 002654        C$RFLA= 000021          EM50H   017652          FMT06   013530
BIT12   = 010000 G        BT2     = 002740        C$RP1 = 000025          EM50I   017740          FMT06A  013542
BIT13   = 020000 G        BUFARE  002654          C$SEFG= 000046          EM50J   020002          FMT07   013476
BIT14   = 040000 G        CFMT0   022524          C$SPRI= 000041          EM50K   020070          FMT09   013621
BIT15   = 100000 G        CFMT2   022613          C$SVEC= 000037          EM50L   020132          FMT09A  013673
BIT2    = 000004 G        CFMT3   022673          C$TPRI= 000013          EM50M   020214          FMT10   013714
BIT3    = 000010 G        CONSOL  002314          DCOMTS= 000014          EM50N   020256          FMT11   013770
BIT4    = 000020 G        CONTIN  022414          DDRA  = 120003          EM50S   020324          FMT12   007672
BIT5    = 000040 G        CONTST  022514          DDRB  = 120002          EM50U   020376          FMT12A  007732
BIT6    = 000100 G        CSREGS= 000010          DELAY1  002262          EM50V   020442          FMT12D  010020
BIT7    = 000200 G        C$AU  = 000052          DELAY2  002264          EM50W   020507          FMT12E  010063
BIT8    = 000400 G        C$AUTO= 000061          DEVMAP  002310          EM50X   020556          FMT12F  010145
BIT9    = 001000 G        C$BRK = 000022          DEVPTR  002312          EM50Y   020621          FMT13A  010546
BOE     = 000400 G        C$BSEG= 000004          DFPTBL  002160 G        EM50Z   020673          FMT13B  010611
BRDTYP  002366            C$BSUB= 000002          DIAGMC= 000000          EM51B   020734          FMT13C  010656
BSEL    002320            C$CEFG= 000045          DMVDAI= 000001          EM51C   021002          FMT13D  010721
BSELRS  002622            C$CLCK= 000062          DMVPU = 000004          EM51E   021050          FMT13E  010764
BSEL0   002320            C$CLEA= 000012          EF.CON= 000036 G        EM51F   021104          FMT16   014007
BSEL1   002322            C$CLOS= 000035          EF.NEW= 000035 G        EM51G   021151          FMT16A  014032
BSEL10  002340            C$CLP1= 000006          EF.PWR= 000034 G        EM51L   021213          FMT17   014074
BSEL11  002342            C$CVEC= 000036          EF.RES= 000037 G        EM51M   021275          FMT17A  014153
BSEL12  002344            C$DCLN= 000044          EF.STA= 000040 G        EM51N   021337          FMT17B  014234
BSEL13  002346            C$DODU= 000051          EMCODE  034754          EM51P   021405          FMT17C  014307
BSEL14  002350            C$DRPT= 000024          EM26    016152          EM52A   021457          FMT4    013213
BSEL15  002352            C$DU  = 000053          EM26A   016204          EM52B   021535          FMT4A   013253
BSEL16  002354            C$EDIT= 000003          EM26B   016226          EM52C   021567          FMT4B   013306
BSEL17  002356            C$ERDF= 000055          EM26C   016251          EM52D   021653          FMT4C   013313
BSEL2   002324            C$ERHR= 000056          EM26D   016274          EM60N   021711          FMT5    013346
BSEL3   002326            C$ERRO= 000060          EM26E   016316          ENDEMB  013120          FMT5A   013411
BSEL4   002330            C$ERSF= 000054          EM26F   016351          ERRBLK  002210 G        FMT50A  014367
BSEL5   002332            C$ERSO= 000057          EM26G   016402          ERRFLG  002276          FMT50B  014441
BSEL6   002334            C$ESCA= 000010          EM27    016424          ERRMSG  002206 G        FMT50C  014522
BSEL7   002336            C$ESEG= 000005          EM27A   016437          ERRNBR  002204 G        FMT50D  014562
BSLT0   = 000020          C$ESUB= 000003          EM27B   016461          ERRTYP  002202 G        FMT50E  014577
BSLT1   = 000021          C$ETST= 000001          EM27C   016502          ERR1    006212 G        FMT50M  014614
BSLT2   = 000022          C$EXIT= 000032          EM3     016115          ERR10   007306 G        FMT51A  014673
```

SYMBOL TABLE

```
FMT52A  011557        HLTST2  035600        L$AUT   002070 G      L10007  007276        MOVLTD  004166
FMT52B  011607        HOE     = 100000 G    L$AUTO  022752 G      L10010  007304        MOVSB   005034
FMT52C  011652        IBE     = 010000 G    L$CCP   002106 G      L10011  007312        MOVSW   005004
FMT52H  011514        IDU     = 000040 G    L$CLEA  023076 G      L10012  007454        MPCSR   002320
FMT60   012034        IENBA   = 000001       L$CO    002032 G      L10013  007670        MPIHAN  006066 G
FMT61   012073        IENBB   = 000020       L$DEPO  002011 G      L10014  010544        MPIVEC  002360
FMT62   012134        IENR    = 120016       L$DESC  003274 G      L10015  011174        MPOHAN  006140 G
FRSPAS  002304        IER     = 020000 G     L$DESP  002076 G      L10016  011266        MPOVEC  002362
FRSTIM  002302        IFR     = 120015       L$DEVP  002060 G      L10017  011512        MPRIOR  002364
F$AU    = 000015      IFRCA1  = 000002       L$DISP  002124 G      L10020  012032        MRDY    = 000200
F$AUTO  = 000020      IFRCA2  = 000001       L$DLY   002116 G      L10022  022512        MREQ    = 000001
F$BGN   = 000040      IFRCB1  = 000020       L$DTP   002040 G      L10023  023066        MSTCLR  003514
F$CLEA  = 000007      IFRCB2  = 000010       L$DTYP  002034 G      L10024  023112        NEWLIN  013124
F$DU    = 000016      IFRIRQ  = 000200       L$DU    023114 G      L10025  023116        NEWPC   002452
F$END   = 000041      IFRSR   = 000004       L$DUT   002072 G      L10026  023120        NEWST   022232
F$HARD  = 000004      IFRT1   = 000100       L$DVTY  003254 G      L10027  024036        NPRPBT  = 000200
F$HW    = 000013      IFRT2   = 000040       L$EF    002052 G      L10030  024034        NPRAIH  = 000075
F$INIT  = 000006      IHILNK  006136         L$ENV1  002044 G      L10031  024756        NPRAIL  = 000074
F$JMP   = 000050      IHOLNK  006210         L$ERRT  002202 G      L10032  025404        NPRAIX  = 000076
F$MOD   = 000000      INITT1  004456         L$ETP   002102 G      L10033  026114        NPRAOH  = 000071
F$MSG   = 000011      INITT2  004630         L$EXP1  002046 G      L10034  025656        NPRAOL  = 000070
F$PROT  = 000021      INTFLG  002272         L$EXP4  002064 G      L10035  026112        NPRAOX  = 000072
F$PWR   = 000017      INTWCH  002274         L$EXP5  002066 G      L10036  026364        NPRBS7  = 000200
F$RPT   = 000012      IRQA    = 000004       L$HARD  036344 G      L10037  026772        NPRBYT  = 000010
F$SEG   = 000003      IRQB    = 000002       L$HIME  002120 G      L10040  026566        NPRCTL  = 123004
F$SOFT  = 000005      IRQREG  = 123005       L$HPCP  002016 G      L10041  026770        NPRDL   = 000044
F$SRV   = 000010      ISR     = 000100 G     L$HPTP  002022 G      L10042  030514        NPRDLB  = 000054
F$SUB   = 000002      IXE     = 004000 G     L$HW    002160 G      L10043  027706        NPRDRH  = 123001
F$SW    = 000014      I$AU    = 000041       L$ICP   002104 G      L10044  030512        NPRDRL  = 123000
F$TEST  = 000001      I$AUTO  = 000041       L$INIT  022020 G      L10045  031232        NPREAD  005134
GDATA   002254        I$CLN   = 000041       L$LADP  002026 G      L10046  032460        NPRGO   = 000100
GETBSR  004232        I$DU    = 000041       L$LAST  037054 G      L10047  031722        NPRIO   = 000040
GETPRM  022256        I$HRD   = 000041       L$LOAD  002100 G      L10050  032456        NPRLD   = 000004
GETSR   024760        I$INIT  = 000041       L$LUN   002074 G      L10051  034040        NPRMCR  002632
GETT2   024072        I$MOD   = 000041       L$MREV  002050 G      L10052  032752        NPRMOV  005266
GETWSR  004374        I$MSG   = 000041       L$NAME  002000 G      L10053  033156        NPROTS  = 000004
G$CNTO  = 000200      I$PROT  = 000040       L$PRIO  002042 G      L10054  033362        NPRTOE  = 000006
G$DELM  = 000372      I$PTAB  = 000041       L$PROT  022012 G      L10055  033566        NULERR  013076
G$DISP  = 000003      I$PWR   = 000041       L$PRT   002112 G      L10056  033710        OLDSP   002454
G$EXCP  = 000400      I$RPT   = 000041       L$REPP  002062 G      L10057  034032        ORA     = 120001
G$HILI  = 000002      I$SEG   = 000041       L$REV   002010 G      L10060  034134        ORAM    = 120017
G$LOLI  = 000001      I$SETU  = 000041       L$SOFT  036742 G      L10061  034200        ORB     = 120000
G$NO    = 000000      I$SFT   = 000041       L$SPC   002056 G      L10062  034712        O$APTS  = 000000
G$OFFS  = 000400      I$SRV   = 000041       L$SPCP  002020 G      L10063  034516        O$AU    = 000001
G$OFSI  = 000376      I$SUB   = 000041       L$SPTP  002024 G      L10064  034710        O$BGNR  = 000000
G$PRMA  = 000001      I$TST   = 000041       L$STA   002030 G      L10065  035760        O$BGNS  = 000000
G$PRMD  = 000002      J$JMP   = 000167       L$SW    002202 G      L10066  035310        O$DU    = 000001
G$PRML  = 000000      LODT2C  024040         L$TEST  002014 G      L10067  035756        O$ERRT  = 000001
G$RADA  = 000140      LOE     = 040000 G     L$TIML  002014 G      L10070  036416        O$GNSW  = 000000
G$RADB  = 000000      LOGDEV  002266         L$UNIT  002012 G      L10071  036742        O$POIN  = 000001
G$RADD  = 000040      LOT     = 000010 G     L10000  002200        MASCLR  003346        O$SETU  = 000000
G$RADL  = 000120      LSIDCL  = 000002       L10001  002202        MCLR    = 000100      PATB    002456
G$RADO  = 000020      LSIHLT  = 000020       L10002  006134        MC1     035762        PATCH   036742
G$XFER  = 000004      LU1MOD  002000 G       L10003  006206        MC2     036076        PATF    002506
G$YES   = 000010      L$ACP   002110 G       L10004  006216        MC2END  036341        PCR     = 120014
HELP    = 000000      L$APT   002036 G       L10005  006230        MFEND   032460        PFLAG   002316
HLTEST  = 000015      L$AU    023120 G       L10006  006530        MLWRI   004064        PNT     = 001000 G
```

SYMBOL TABLE

```
PRI     = 002000 G      STARST  022120      TXT3    015162      T1      023122 G      WRITEI  004054
PRIRTY  036475          SVCGBL= 000000      TXT4    015212      T1CH    = 120005      WSR0    002212
PRI00   = 000000 G      SVCINS= 000001      TXT4A   015252      T1CL    = 120004      WSR10   002222
PRI01   = 000040 G      SVCSUB= 000001      TXT6    015313      T1LH    = 120007      WSR12   002224
PRI02   = 000100 G      SVCTAG= 000001      TXT8A   015336      T1LHGO= 120005        WSR14   002226
PRI03   = 000140 G      SVCTST= 000001      TXT8B   015353      T1LL    = 120006      WSR16   002230
PRI04   = 000200 G      SWPBOT= 121000      TXT8C   015370      T1.1    023122        WSR2    002214
PRI05   = 000240 G      SWPDDC= 121400      TXT8D   015405      T10     034042 G      WSR4    002216
PRI06   = 000300 G      S$LSYM= 010000      TXT8E   015422      T11     034136 G      WSR6    002220
PRI07   = 000340 G      TDATA   002252      T$ARGC= 000002      T12     034202 G      W0      = 003054
PSTACK  002270          TMPA    002436      T$CODE= 007130      T12.1   034262        W1      = 003056
PT.CTL  002370          TMPB    002440      T$ERRN= 000126      T12.2   034520        W2      = 003060
PU24    = 000001        TMPC    002442      T$EXCP= 000000      T13     034754 G      W3      = 003062
PU24.M  036611          TMPD    002444      T$FLAG= 000040      T13.1   035034        W4      = 003064
READ    003616          TMPE    002446      T$GMAN= 000000      T13.2   035312        W5      = 003066
READI   003730          TMPF    002450      T$HILI= 000002      T2      024132 G      W6      = 003070
REDLOC= 000001          TMP0    002412      T$LAST= 000001      T2CH    = 120011      W7      = 003072
REDPAG= 000003          TMP1    002414      T$LOLI= 000000      T2CL    = 120010      W8      = 003074
REGNUM  002300          TMP2    002416      T$LSYM= 010000      T2LL    = 120010      W9      = 003076
REG0    002372          TMP3    002420      T$LTNO= 000015      T3      025020 G      XDATA   002260
REG1    002374          TMP4    002422      T$NEST= 177777      T4      025406 G      XLOC0   032462
REG2    002376          TMP5    002424      T$NS0 = 000000      T4.1    025422        XLOC1   032464
REG3    002400          TMP6    002426      T$NS1 = 000005      T4.2    025660        XLOC2   032466
REG4    002402          TMP7    002430      T$NS2 = 000002      T5      026116 G      XLOC3   032470
REG5    002404          TMP8    002432      T$PTNU= 000000      T6      026366 G      XLOC4   032472
REG6    002406          TMP9    002434      T$SAVL= 177777      T6.1    026366        XLOC5   032474
REG7    002410          TXTMLT  021746      T$SEGL= 177777      T6.2    026570        XLOC6   032476
RESTRT  022222          TXTML0  015515      T$SUBN= 000002      T7      026774 G      XMFG.M  036673
RUN     = 000200        TXTML1  015521      T$TAGL= 177777      T7.1    027032        XMINIT  030516
RXVAL0  032516          TXTML2  015535      T$TAGN= 010072      T7.2    027710        XMINTH  031062 G
RXVAL1  032520          TXTML3  015552      T$TEMP= 000000      T8      031234 G      XMREAD  030734
RXVAL2  032522          TXTML4  015574      T$TEST= 000015      T8LP    031734        XMWRIT  030636
RXVAL3  032524          TXTML5  015615      T$TSTM= 177777      T8.1    031264        XM4HOL  031060
RXVAL4  032526          TXTML6  015645      T$TSTS= 000001      T8.2    031724        XM4INT  031032
RXVAL5  032530          TXTML7  015657      T$$AU = 010026      T9      032534 G      XORGB   012162
RXVAL6  032532          TXTNP   015745      T$$AUT= 010023      T9.1    032550        XORSW   005064
SEL0    002320          TXTNPT  021770      T$$CLE= 010024      T9.2    032754        XVAL0   032500
SEL10   002340          TXTNP0  015752      T$$DU = 010025      T9.3    033160        XVAL1   032502
SEL12   002344          TXTNP1  015762      T$$HAR= 010070      T9.4    033364        XVAL2   032504
SEL14   002350          TXTNP2  015772      T$$HW = 010000      T9.5    033570        XVAL3   032506
SEL16   002354          TXTNP3  016002      T$$INI= 010022      T9.6    033712        XVAL4   032510
SEL2    002324          TXTNP4  016017      T$$MSG= 010020      UAM     = 000200 G    XVAL5   032512
SEL4    002330          TXTNP5  016034      T$$PRO= 010021      VECTOR  036444        XVAL6   032514
SEL6    002334          TXTNP6  016051      T$$SOF= 010071      WA      = 003100      X$ALWA= 000000
SFPTBL  002202 G        TXTNP7  016065      T$$SRV= 010045      WB      = 003102      X$FALS= 000040
SLT0    = 000020        TXTNP8  016101      T$$SUB= 010067      WC      = 003104      X$OFFS= 000400
SLT2    = 000022        TXTNUL  015513      T$$SW = 010001      WD      = 003106      X$TRUE= 000020
SLT4    = 000024        TXT1    014761      T$$TES= 010065      WE      = 003110      $F      = 000126
SLT6    = 000026        TXT11A  015437      T.EDF = 000001      WF      = 003112      $LSTIN= 000001
SMCODE  034714          TXT11B  015460      T.EHRD= 000002      WRILOC= 000002        $LSTTA= 000001
SR      = 120012        TXT2    015017      T.ESF = 000000      WRIPAG= 000004        $MPCSR= 160000 G
STALL   005132          TXT2A   015061      T.ESFT= 000003      WRITE   004042        $T      = 000015
STARES  002306          TXT2B   015120

. ABS.  037054    000
        000000    001
ERRORS DETECTED:  0
```

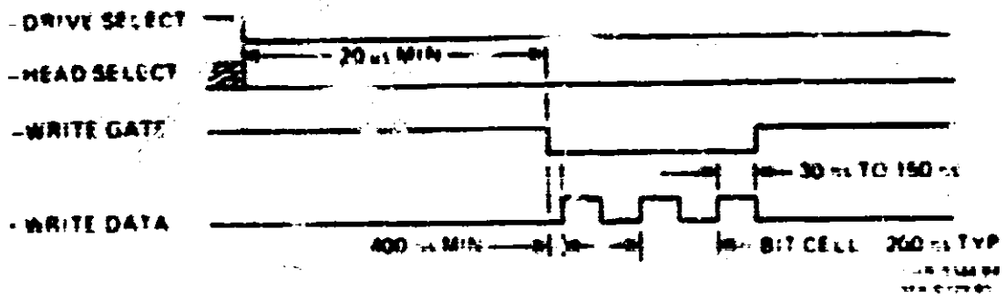


Figure 4-11 MFM Write Data Timing

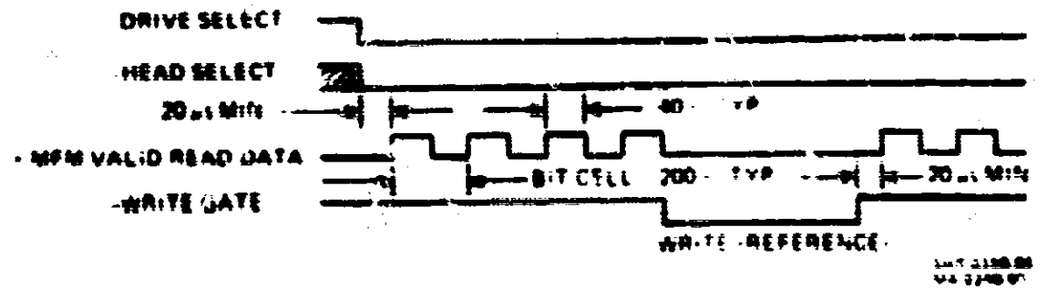


Figure 4-12 MFM Read Data Timing

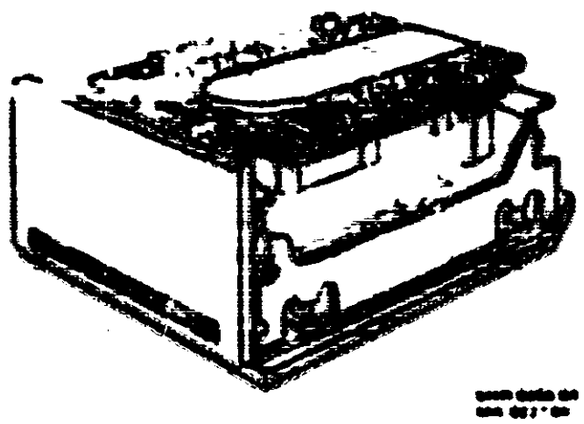


Figure 4-13 The RD52-A Disk Drive (MK-2122-4120)

Mechanical Dimensions

Height	[illegible] cm ([illegible] in)
Width	14.6 cm (5.75 in)
Depth	20.32 cm (8.00 in)
Weight	1.15 kg (2 lb)

Heat Dissipation

[illegible] W (119 Btu/hr) typical
41.5 W (148 Btu/hr) maximum

Shock and Vibration

	Operating	Nonoperating
Shock	11 G's and 2 msec 0.8 G's Y axis	60 G's
Vibration	0.5 G's at 10 - 500 Hz 0.1 inch at 2 - 10 Hz	2 G's at 10 - 500 Hz 0.4 inch at 2 - 10 Hz

9.10.2.2 Performance Specifications

Capacity (unformatted)

Per drive (megabytes)	47.02
Per surface (megabytes)	6.61
Per Track (bytes)	10,416

Access Times
(including head settling)

Track to track	6 ms
Average	33 ms
Full stroke	60 ms
Average latency	8.33 ms

9.10.2.3 Functional Specifications

Nominal rotating speed (rpm)	[illegible]
Maximum rotating speed (rpm)	[illegible]
Minimum rotating speed (rpm)	[illegible]
Maximum recording density (bits/in)	[illegible]
Maximum flux density (flux changes/in)	[illegible]
Track density (tracks/in)	[illegible]
Cylinders	[illegible]
Tracks	[illegible]
Read/write heads	7
Disks	4
Index	1

9.10.3 Power Requirements

9.10.3.1 IK Power - The voltages required to operate the drive are +5 Vdc and +12 Vdc. Figures 9-14 and 9-15 show the current requirements in more detail.

9.10.3.2 IK Power Connector - The dc power connector (J1) is a 4-pin AMP Mate-N-Lock connector (AMP P/N 1-480424-0) on the component side of the bottom PCB. You can access the connector from the back of the drive. Digital recommends that you make the dc power connector with AMP P/N 1-480424-0. Figure 9-16 shows the dc power connector pins.

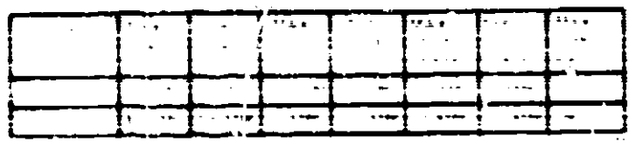


Figure 9-14 Current Requirements

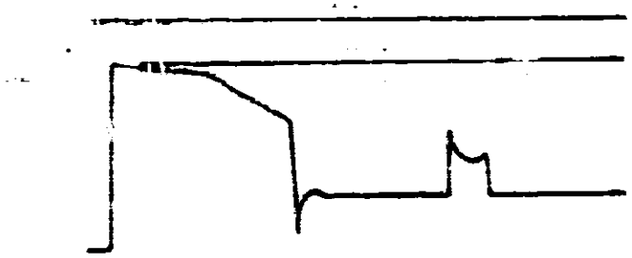


Figure 9-15 +12 V Startup Current

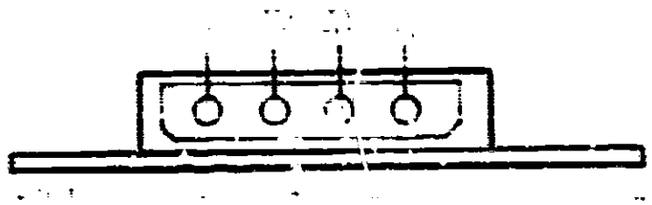


Figure 9-16 J1 Connector Layout

FOR ENLARGED ART
PLEASE SEE FOLLOWING
FRAME(S).

VOLTAGE	MAX START	TYP START	MAX SEEK ING	TYP SEEK ING	MAX STEADY STATE	TYP STEADY STATE	MAX RIPPLE P-P
+5	1.5 AMP	1 AMP	1.5 AMP	1 AMP	1.5 AMP	1 AMP	50 mV
+12	4.5 AMP	3.5 AMP	3 AMP	2.5 AMP	2 AMP	1.5 AMP	80 mV

Figure 4-13 Current Requirements

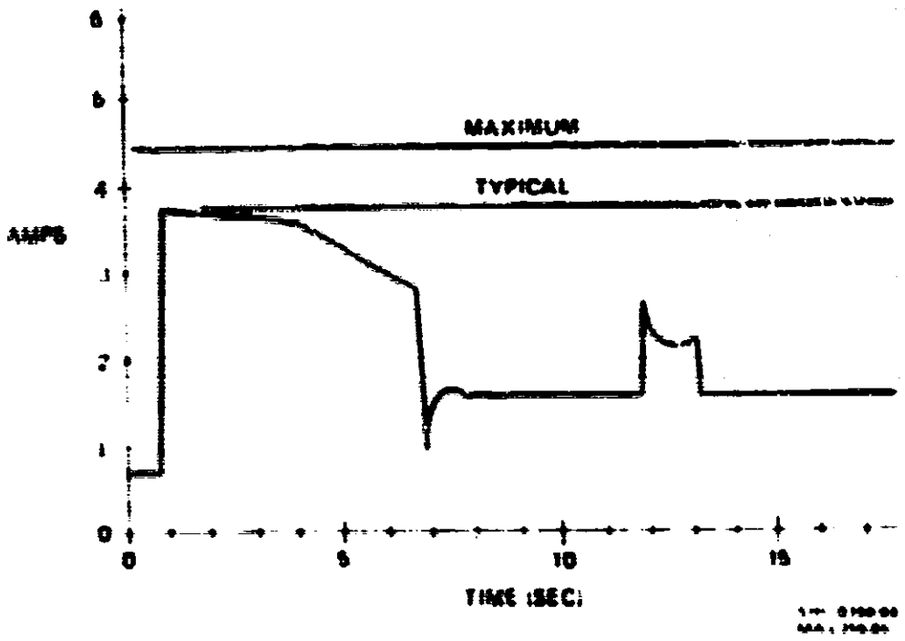
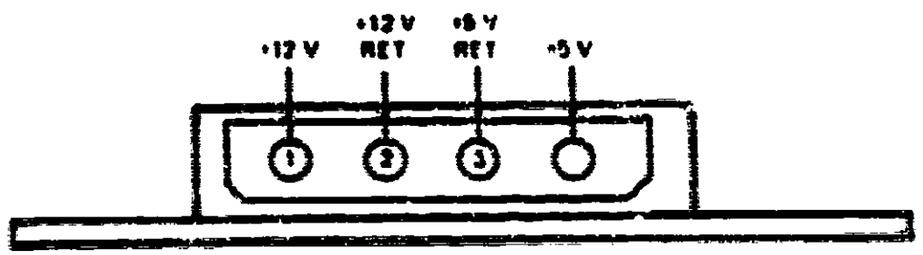


Figure 4-14 +12 V Startup Current



NOTE This is the drive end of the connector

Figure 4-15 J1 Connector Layout

9.11 PHYSICAL DESCRIPTION

9.11.1 Assemblies

The RD52-A disk drive has the following major assemblies:

Read/write interface PCB
Spindle/EMA drive PCB
Servo control PCB
Spindle drive mechanism
Positioning mechanism
Read/write heads and media

These assemblies are shown in Figure 9-17 and described in the following paragraphs.

9.11.1.1 Read/Write Interface PCB - The read/write interface PCB, to which all power, control and data signals are connected, provides the following functions:

- Receives power and regulates internal voltage
- Controls INPUT signal reception and internal distribution
- Controls OUTPUT signal accumulation, sequencing, and transmission
- Controls READ/WRITE signal bidirectional reception, conditioning and transmission
- Detects fault and generates fault signals

9.11.1.2 Spindle/EMA Drive PCB - The spindle/EMA drive PCB has a dedicated microcomputer and is mounted to the bottom of the sealed mechanical enclosure. This PCB derives its power from the read/write interface PCB and provides the following functions:

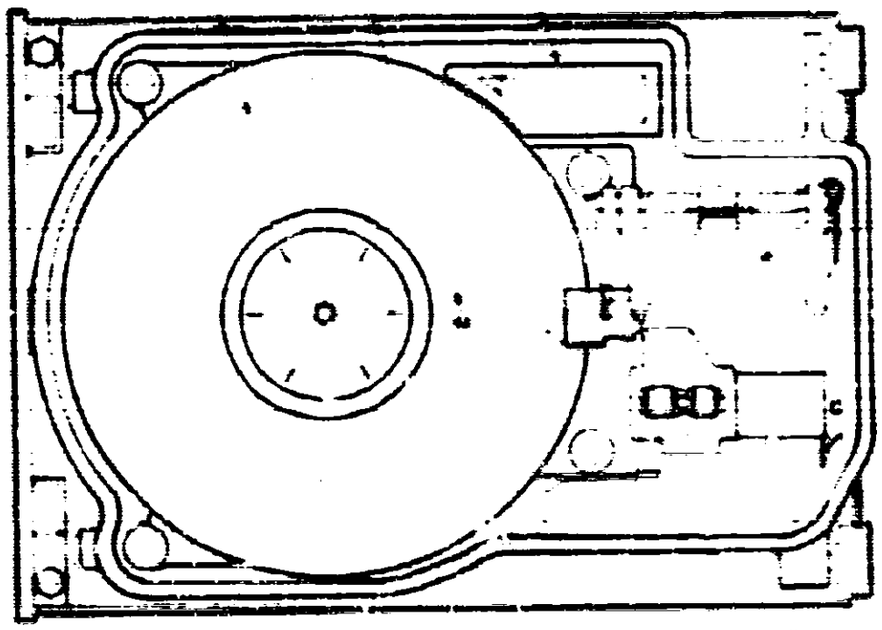
Power and speed control to the spindle drive motor
Power drive to the electromagnetic actuator (EMA)

9.11.1.3 Servo Control PCB – The servo control PCB has a dedicated microcomputer and is mounted to the top cover of the sealed mechanical enclosure. The servo control PCB provides the following functions:

- Controls and monitors signal sequencing during the power-up operation
- Receives the servo data that is read from the dedicated servo disk surface by the servo head
- Conditions the servo data and generates position signals
- Distinguishes between single and buffered step mode seeks, then for buffered step mode seeks, generates, detects, and controls the carriage velocity to ensure arrival at the desired cylinder
- Controls continuous position while on track

9.11.1.4 Spindle Drive Mechanism – A brushless dc spindle drive motor rotates the spindle at 3600 rpm (±1 percent). The motor is thermally isolated from the baseplate to minimize temperature transfer. The motor, spindle, and disk stack are dynamically balanced to eliminate vibration. A dedicated microcomputer provides complete control over the spindle rotation, and allows algorithm control of the motor start and stop.

9.11.1.5 Positioning Mechanism – The read/write heads are mounted on a ballbearing-supported linear carriage. The carriage is positioned by a linear voice coil motor and driven by the closed loop servo system.



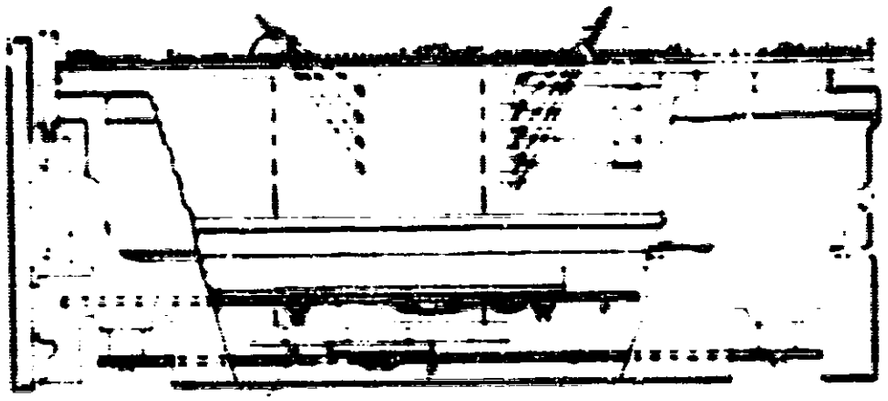
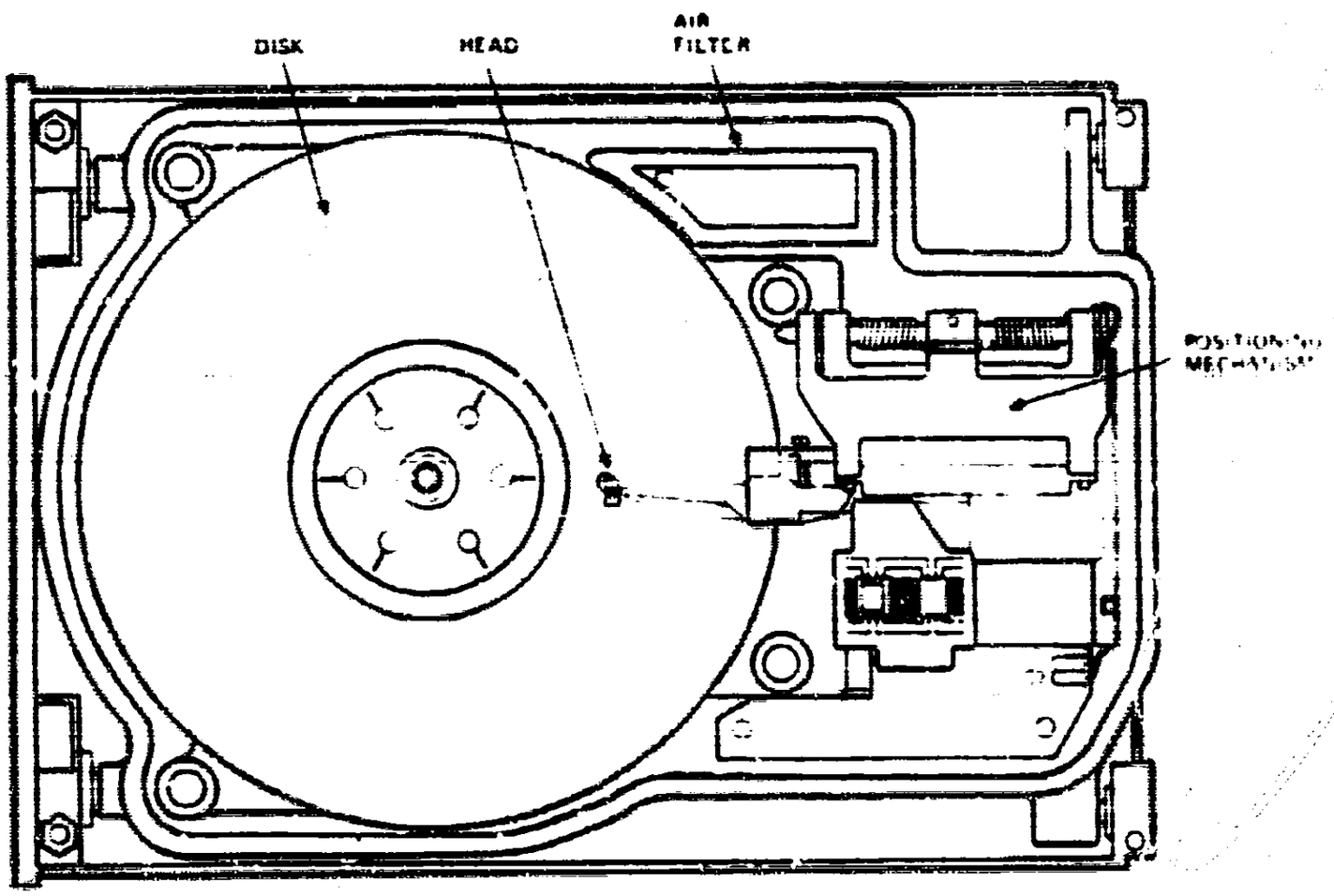


Figure 9-17 RD52-A Major Assemblies

FOR ENLARGED ART
PLEASE SEE FOLLOWING
FRAME(S).


DISK
HEAD
AIR FILTER

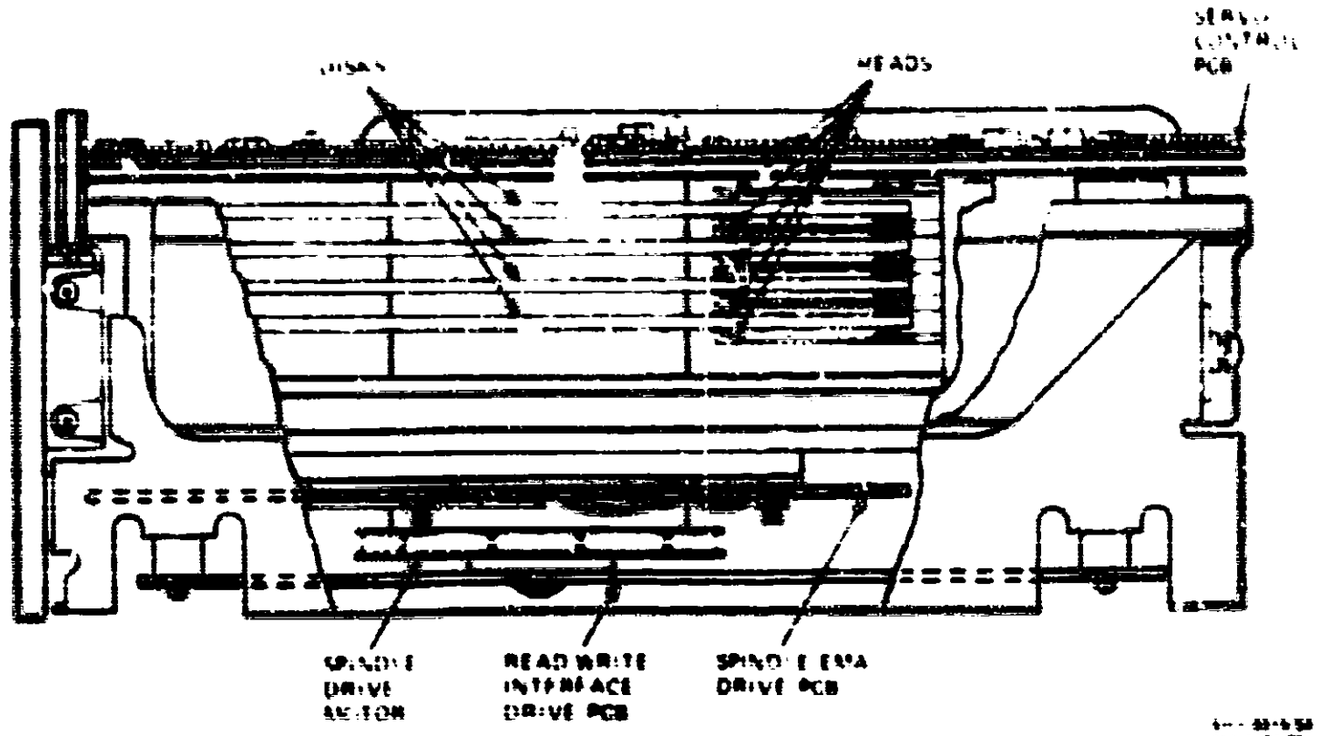

HEADS
SPINDLE DRIVE MOTOR
READ WRITE INTERFACE DRIVE PCB
SPINDLE DRIVE PCB

Figure 4-17 [illegible]

9.11.1.6 Read/Write Heads and Media - The recording media consists of a lubricated thin magnetic oxide coating on a 130 mm diameter aluminum substrate. This coating formulation with the low load force, low mass Winchester type heads, ensure reliable contact start stop operations. Data on each of the eight disk surfaces (four surfaces per disk) is read or written by one read/write head. Each head accesses 843 data cylinders.

9.11.2 Air Filtration System
The disks and read/write heads are enclosed in a contamination-free mechanical enclosure. The RD53-A uses an integral recirculating air system with an absolute filter (Figure 9-37) to maintain a clean environment. The filter has a port that permits pressure equalization with the ambient air.

9.12 OPERATION

9.12.1 Operator Functions
Operator functions vary depending on the system in which the RD53-A is installed. The information in this subsection describes how the operator can select drives and terminate the control cable. Figure 9-38 shows the location of the drive select jumpers and the terminator on the read/write interface PCB.

9.12.1.1 Drive Select - Five jumpers are provided for logical drive number assignment. Drive select 1, 2, 3, and 4 cause the drive to be selected when the active drive select line matches the installed number. When installed, jumper R (default) causes the drive to be selected constantly. The R jumper option is never used in Digital applications.

9.12.1.2 Control Cable Termination - If the RD53-A is the last drive at the end of the control signal cable, a 220 ohm terminator resistor pack must be installed. This termination must be removed if the drive is not the last in a string of drives. In Digital applications, the terminator pack is always installed.

Figure 9-38 Jumper/Terminator Location

9.12.2 Drive Organization
Figure 9-39 is a block diagram of the RD53-A. Paragraphs 9.12.3 through 9.12.4 describe RD53-A operation.

9.12.3 Power Sequencing
The +5 Vdc and +12 Vdc power can be applied in any order. The +12 Vdc must be applied to start the spindle drive motor. A microcomputer monitors the disk rotation. When the disks are spinning at 3600 rpm (+1 percent), the heads automatically recalibrate to track 0. When the heads arrive at track 0, the TRACK 000 signal is activated. The TRACK 000 signal precedes the READY and SEEK COMPLETE signals by less than 90 ms. The READY signal is not activated if there is a fault condition. The drive can perform read, write, or seek operations only if the READY signal is active.

9.12.4 Microprocessor Wake-Up Timing

9.12.4.1 Spindle Open Loop Acceleration Ramp - The RD53-A acceleration time in normal is about 10 seconds. The spindle reaches its current limit, which is 1.5 A rms (Figure 9-40).

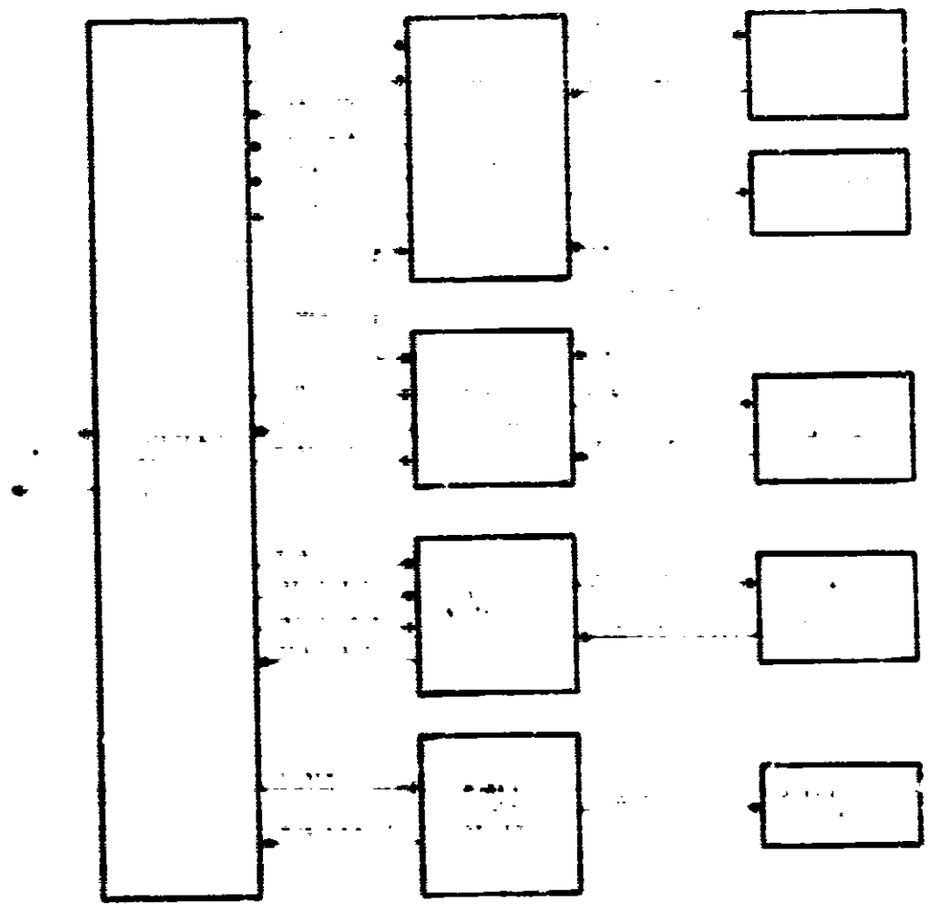


Figure 9-39 RD53-A Block Diagram

FOR ENLARGED ART
PLEASE SEE FOLLOWING
FRAME(S).

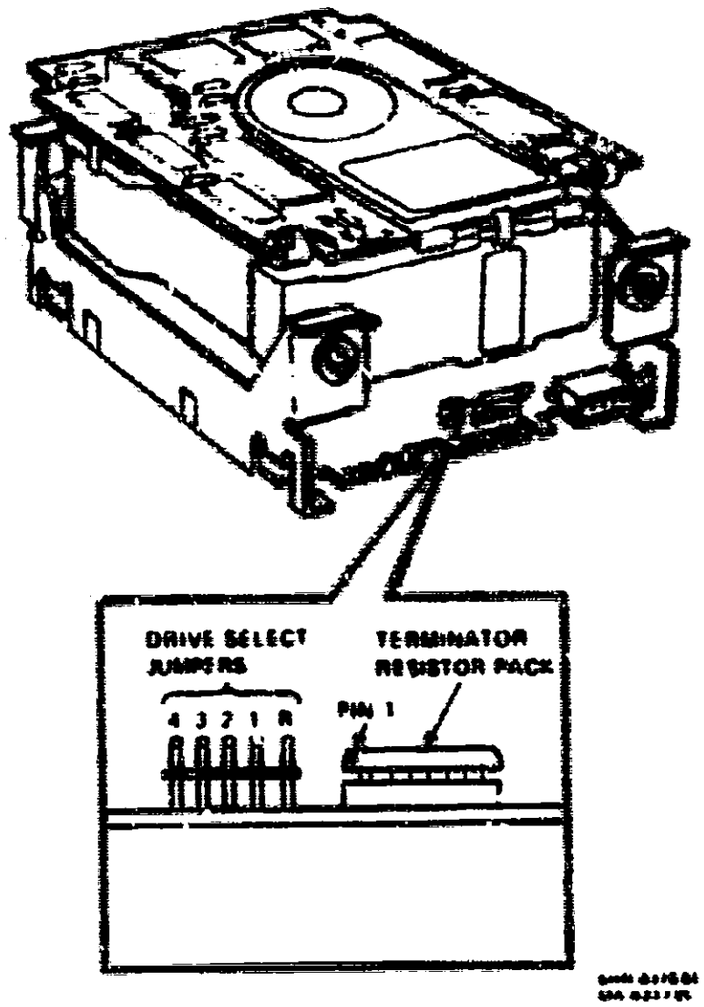


Figure 9-38 Jumper/Terminator Locations

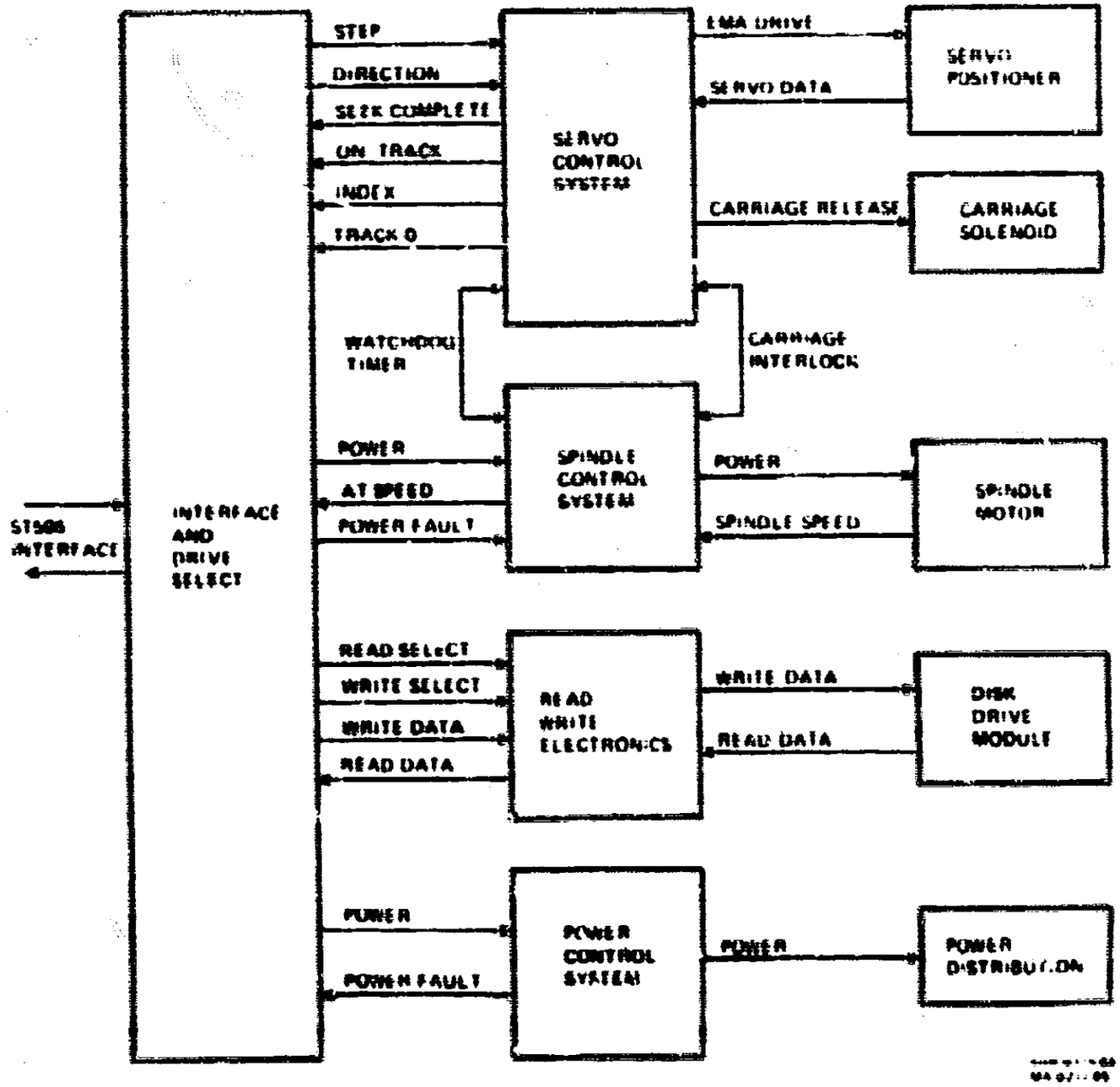


Figure 9-39 [illegible] Block Diagram

FOR ENLARGED ART
PLEASE SEE FOLLOWING
FRAME(S).

9.12.4.2 Spindle Driver - The spindle driver is a unipolar, 3-phase drive. The motor windings are connected in a Y configuration with the centertap connected to +12 Vdc.

9.12.4.3 Servo Control of the Spindle - Servo control of the spindle speed is done by integrating the speed error over time and applying a correction factor to the spindle speed to keep the speed constant.

A timer initializes at the same point on each revolution. Since the timer runs independent of the program that runs the motor, it is used as a speed-checking device. If the timer is not at the correct value, the speed of the spindle is too fast or too slow.

A first check is made to see if the timer is within a certain range of values. If it is out of range, then a gross adjustment is made to the pulse size to correct the condition (the pulse width is set to maximum or minimum according to the condition). If the timer is within the correct range, the value is stored for later use. The value is checked to find out if the speed correction is positive (speed up) or negative (slow down).

The speed correction is done in the integrator part of the software.

The integrator does a repetitive summation of spindle-speed errors over time. The integrator works as follows:

1. If the speed error is a positive value and the previous value is positive, the two values are added and a positive limit check is done. If the sum is greater than the maximum, the maximum value is used.
2. If the speed error is positive and the previous value is negative, or the speed error is negative and the previous error is positive, the two values are added. The sum is the value used for the next revolution.
3. If the speed error is negative and the previous value is negative, the two values are added. The result is checked to see that the sum is not less than the minimum value. If the sum is less than the minimum, the minimum value is used.

The integrator causes the spindle to seek to hold rpm (the speed error value is 0) at the speed. The actual speed is determined by the accuracy of the microcomputer crystal and the resolution of the timer.

The spindle motor does not report AT SPEED until the spindle enters the servo state and goes through 256 revolutions.

The spindle microcomputer has only one interrupt. The interrupt is related to the watch-dog inhibitor.

9.12.4.4 Watch-Dog Inhibitor - The watch-dog inhibitor is a safety device that monitors communications between the servo microcomputer and the spindle microcomputer. Whenever a watch-dog pulse is received from the servo processor, the spindle processor is interrupted. The interrupt subroutine in the spindle processor reloads a register with a predefined value. This register is decremented with each spindle revolution. If the register is decremented to zero before it gets reloaded, a watch-dog time-out pulse is sent back to the servo processor, which causes the servo processor to restart.

The watch-dog software in the spindle processor acts like a one-shot to the servo processor. If the servo processor hangs up, it cannot send watch-dog pulses to the spindle processor. The watch-dog time-out then causes the servo processor to restart.

9.12.4.5 Actuator Disabling - The electromagnetic actuator (VCM) can be disabled by three different methods:

- Servo position circuits
- Output line on the spindle microcomputer
- Circuitry on the spindle VCM PCB

Whenever it leaves the servo state, the spindle microcomputer disables the VCM. The circuitry on the spindle VCM PCB also disables the VCM whenever a loss of +12 Vdc occurs on that PCB. When the +12 V loss occurs, the VCM is disabled and VCM current is removed. The carriage is then retracted and latched, and read/write heads power down.

9.12.4.6 Carriage Latch Release - Applying +24 V to the carriage latch releases the latch. The latch release can be exhibited by pulling low the carriage latch control signal (-LATCH).

The high voltage necessary for the carriage latch (+24 Vdc) is produced by using the back electromagnetic force (BEMF) of the spindle motor, which is then rectified.

Carriage release can only occur when the motor is up to speed (-VSPEED line is set). Once this line is set, the servo processor unlatches the carriage. When the -VSPEED line goes low, the carriage is retracted and latched again.

9.12.4.7 Automatic Recover Routine - After the spindle comes up to speed, the initialization process calls up the automatic recalibrate routine to place the heads on track 0.

9.12.5 Fault Finding

9.12.5.1 Error Detection - Five error conditions exist for the spindle microcomputer to note safety concerns for the drive. By indicating the presence of these conditions, the spindle microcomputer guards against possible power transistor failure and head-to-disk wear. The error codes are listed as follows:

Error Number	Definition (When Invoked)
1	All sensors are low at the same time (start and servo states)
2	All sensors are high at the same time (start and servo states)
3	Four seconds or more elapse for each revolution (start state)
4	Servo state was not achieved within 512 revolutions (start state)
5	Spindle speed is not held +/- 1 percent rpm (servo state)

9.12.5.2 Power Fault Status Inhibitor - When the spindle is in the servo state and the carriage is released from the locked position, the spindle microcomputer checks for power faults twice per revolution. If there is a power fault on the first check, the spindle microcomputer sets a flag. If there is a power fault on the second check and a flag was set on the first, the motor is turned off and the drive stops. The drive remains stopped as long as there is a power fault and for another 30 seconds after a recovery.

If the flag is set on the first check and no power fault is detected during the second check, the flag is reset. If a power fault is only detected on the second check, nothing is done. This procedure effectively debounces the power fault line.

The spindle microcomputer handles brown-outs in two ways. First, a 5 V regulator is used on the 12 V line. Second, whenever the voltage to the processor reaches a point where an incorrect reset might occur, an active device forces a reset to the spindle microcomputer.

9.12.5.3. Power Faults - This paragraph describes how the system identifies power-related problems, and the messages that are generated when these problems occur. Figure 9-40 is a block diagram of the RD52-A power fault circuitry.

The power fault circuitry works through WRITE LOGIC, READY LOGIC, and the spindle microprocessor, and generates WRITE FAULT, READY, and SEEK COMPLETE messages, respectively. These processes are described in more detail as follows.

Power Sense - The presence or absence of power is sensed by a comparator circuit that uses a 1 percent reference and 1 percent resistors.

The normal trip points are as follows.

10.53	12 V low
11.72	12 V high
4.51	5 V low
7.90	8.6 V low (internal supply)
9.26	8.6 V high (internal supply)

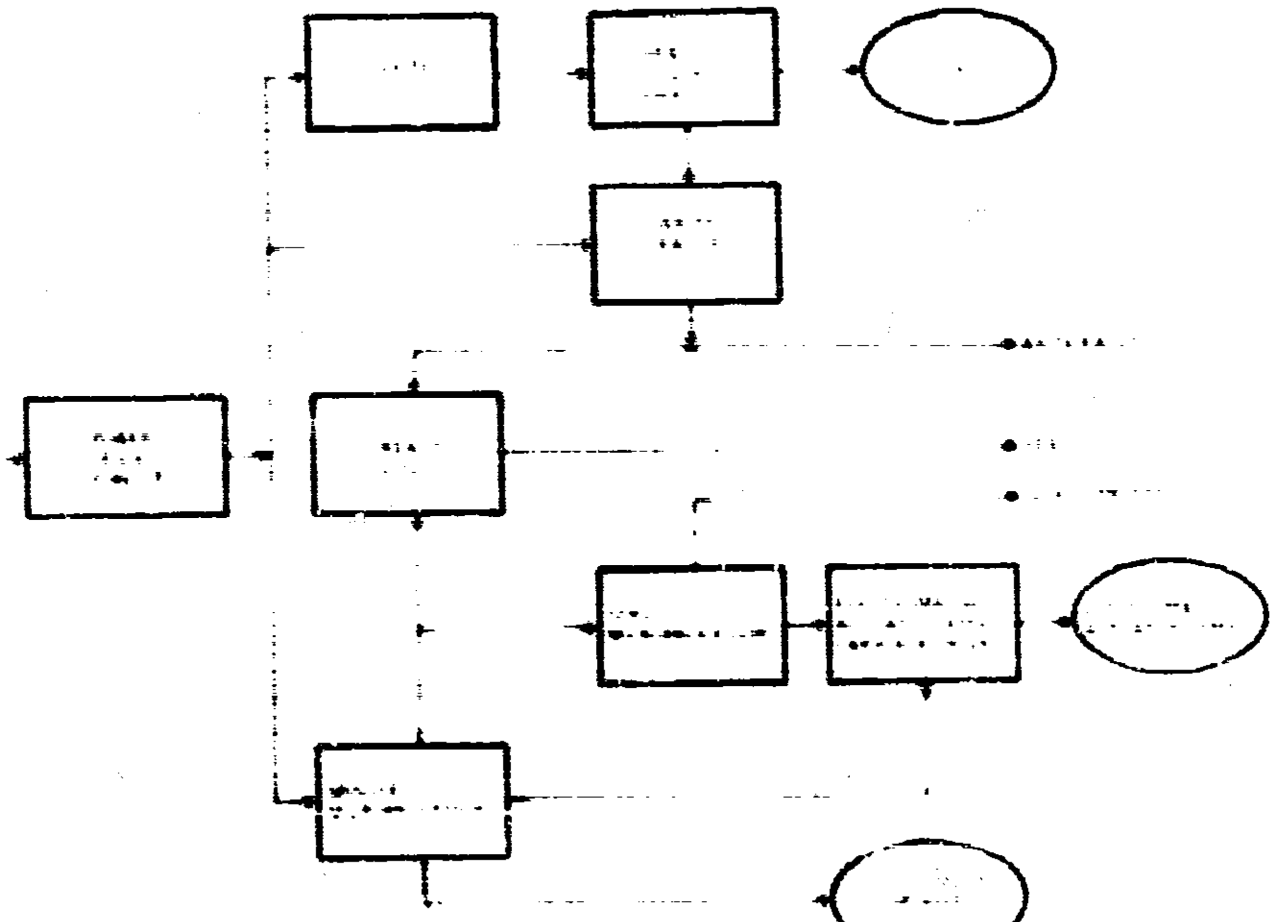


Figure 9-40 Power Faults

Write Logic - Power fault logic prevents writing when power fails. This is accomplished via the write logic block. WRITE CURRENT and the WRITE SELECT are both directly disabled by the power fault circuitry. The not ready condition also inhibits both WRITE SELECT and WRITE DATA.

Ready Logic - The power fault circuitry feeds directly to the ready logic and looks for three conditions: power fault, write fault, or servo fault. A power fault that interacts with the ready logic shows up on the interface as NOT READY.

Write Fault - When power is resumed after a power fault, WRITE FAULT is reset. A WRITE FAULT signal exists only in the true condition.

Spindle Microprocessor - The spindle microprocessor looks at the power fault line twice per revolution. If a power fault is seen twice per revolution, the spindle microprocessor retracts the carriage, spins down, and issues a NOT AT SPEED signal. NOT AT SPEED sends the servo microprocessor to its wake up ... and appears as a servo fault to the ready logic. The ready logic then sends NOT READY to the interface.

Abnormal Conditions - There are two known abnormal conditions. The first condition occurs when the 5 V regulator fails but 12 V is present. This condition causes the spindle to stop and then restart. This happens because the spindle works on 12 V only and the power fault line is invalid with 5 V dead. The second condition occurs if there is a short power fault during a write operation. In this case, the write operation is turned off and READY goes to NOT READY. As soon as the power condition is restored, writing is resumed and READY goes to true.

When the power fault line becomes true, the system shuts down. There are two ways to apply braking to the spindle motor. First, reversing the commutation (applying reverse torque in the stop state) brakes the motor. Second, braking can be initiated by electronically shorting one of the motor phases. (This only occurs if power is removed from the motor circuitry.)

9.12.6 Interface Operations

The interface contains all of the circuits that link the drive to the controller and the drive subsystems to each other. It is composed of many separate logic elements that are best described individually (Figure 9-41).

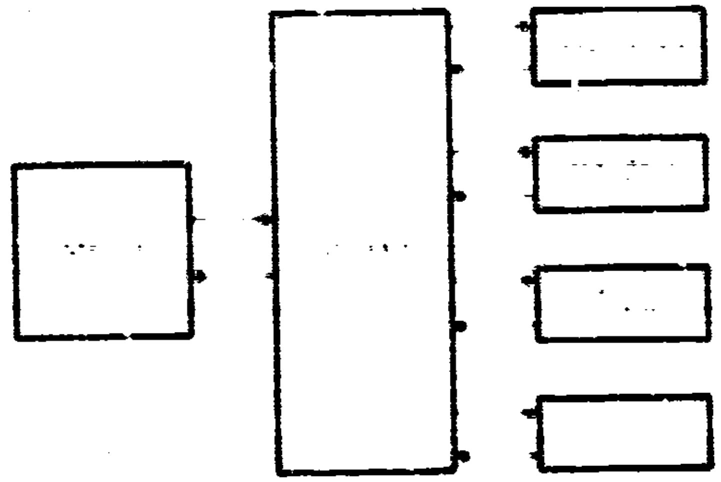


Figure 9-41 Interface Logic

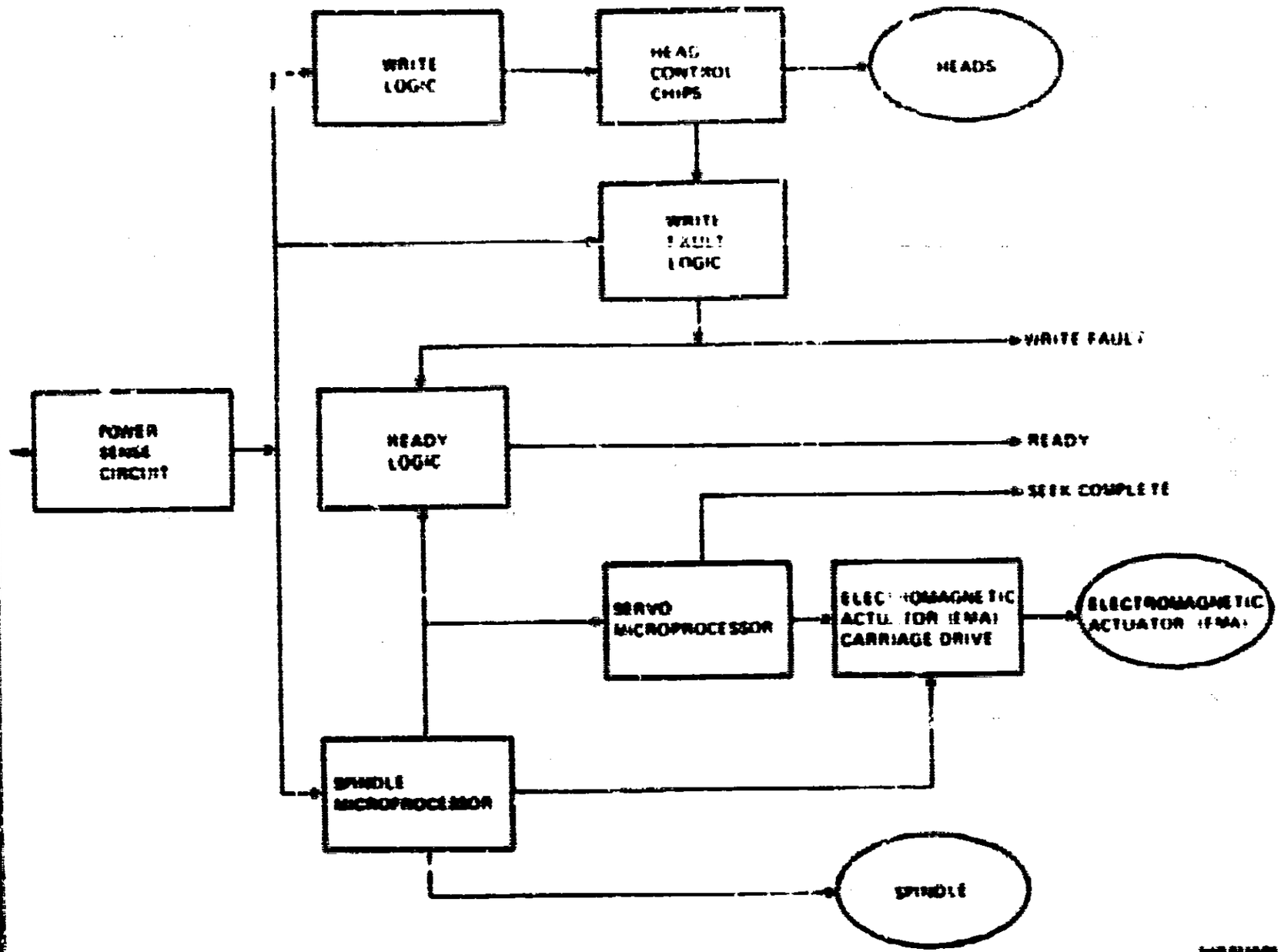


Figure [illegible] Power Fault

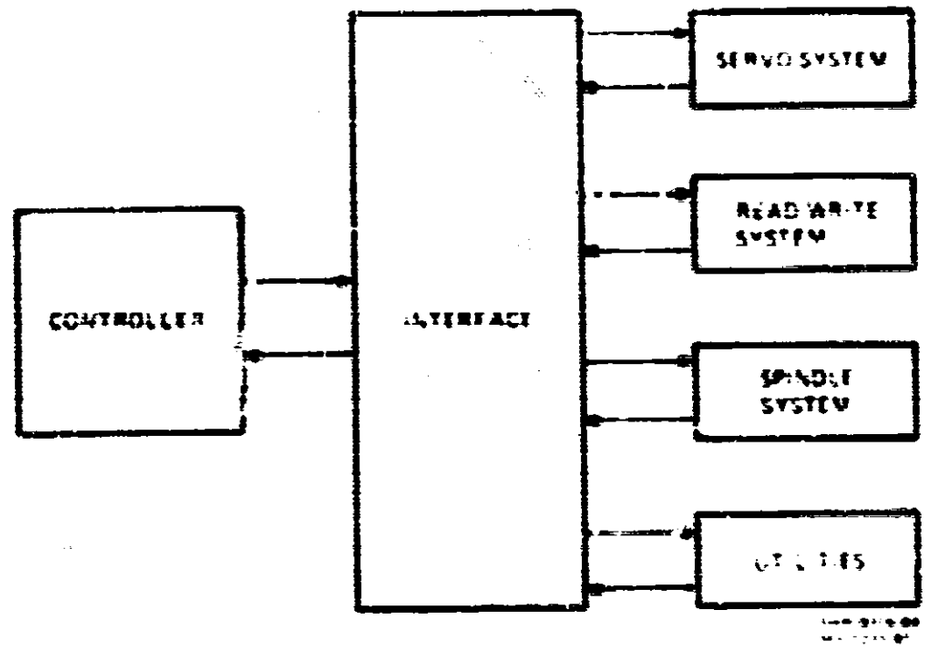


Figure 9-41 Interface Logic

The following lines link the illustrated sections

Power – When power is supplied, the spindle spins up to speed and powers AT SPEED to the servo system and interface. Several system voltages are generated internally: +8, +6, +4, +2, +1, +X

DRIVE SELECT – The user selects the drive in one of two modes: DRIVE SELECT and RADIAL

In RADIAL mode, the drive is selected constantly, enabling communication on the controller cables. The –DRIVE SELECTED signal is fed back to the controller.

In –DRIVE SELECT mode, the drive is selected when the controller selects a jumpered select line (see Paragraph 9.12.1). The drive that is selected enables communication on the controller cables and –DRIVE SELECTED is fed back to the computer.

–TRACK 000 – After power is applied and the spindle is up to speed, the servo does seek to track 0. Track 0 information is written onto the servo track. When the servo is at track 0, it sends a –TRACK 000 signal to the controller module that is enabled by –DRIVE SELECT.

–INDEX – When the servo system is on-track, the interface detects an –INDEX signal and sends it to the index one-shot. The –INDEX signal is written onto the servo surface. The index one-shot signal (200 ms) is gated with –DRIVE SELECT and passed to the controller module.

–DIRECTION IN – The –DIRECTION IN line passes through a latch before it reaches the servo system. The latch is enabled when –STEP is active. When –STEP is inactive (high on the interface), the –DIRECTION IN line remains in its previous state. The –DIRECTION IN line is polled by the servo microprocessor after a step sequence.

–HEAD SELECT – The three head-select lines from the controller module are decoded along with the jumper option (factory set) for the four disk RD5_-A version. The head-select lines are gated with the DRIVE SELECT signal before they pass on to the voltage translators and 104 read/write chip.

–READY – When a drive is selected, the ready line is activated if no faults are present. The following faults prevent –READY from being activated:

Spindle not at speed
–WRITE FAULT
Power fault

–STEP – The –STEP signal must be qualified before it can influence the drive. The following conditions qualify –STEP.

–DRIVE SELECTED
No –WRITE FAULT
No servo fault
No power fault
No –WRITE GATE
–STEP

When qualified, the signal passes to the servo system where the steps are counted. Also the qualified –STEP signal strobes the direction in latch and fires a one-shot, which is logically OR'ed with the –SEEK COMPLETE condition. This produces an early –SEEK-NOT-COMPLETE signal to cover for the servo processor's slow response time.

–SEEK COMPLETE – The –SEEK COMPLETE signal is sent true to a drive when the servo processor has completed a seek and settled on a track. It is sent not true when a step pulse is sent to the drive. The servo processor latches the –SEEK NOT TRUE condition when it determines that a step command has occurred. For –SEEK COMPLETE to appear on the controller interface, the –DRIVE SELECTED condition must be true.

9.12.7 Interface Logic

The –SEEK COMPLETE signal also influences the write process. The –SEEK COMPLETE signal enables WRITE SELECT and WRITE DATA to these pins for the read/write circuitry.

WRITE SELECT – The WRITE SELECT mode is selected if the following conditions exist:

The drive is ready (no faults)
–SEEK COMPLETE is true
The drive is selected
WRITE GATE is active (from the controller)

WRITE DATA – WRITE DATA passed if the following signals are all true:

–SEEK COMPLETE
–DRIVE SELECTED
WRITE GATE
READ

WRITE CURRENT – WRITE CURRENT is gated identically to WRITE SELECT, with the addition of a WRITE BOOST signal, which is added when the servo is on the outer tracks.

–WRITE FAULT – WRITE FAULT is a signal captured in a latch, which can be set by an unsafe signal coming from the 104 read/write control chip. The chip produces an unsafe signal under the following conditions:

WRITE CURRENT without WRITE DATA
An open or short head with an attempt to write
WRITE CURRENT in read mode
For several microseconds after the start of a write operation

The 104 FAULT signal is filtered to eliminate the microsecond fault glitch, and then passed onto the WRITE FAULT latch.

The WRITE FAULT latch can be reset by either a power fault or a false (high) –DRIVE NOT SELECTED.

The –WRITE FAULT condition also feeds back into the interface, where it inhibits steps and produces NOT READY. The NOT READY condition in turn gates both WRITE SELECT and WRITE DATA. A WRITE FAULT inhibits further writing or step acquisition until it is cleared by a power-down or by a drive deselect.

READ DATA – READ DATA is gated to the controller if the drive is selected and –WRITE GATE active is not being sent to the drive.

APPENDIX – A

APPENDIX A
DIAGNOSTIC, ERROR, AND DEVICE CODES

The following table lists the diagnostic errors that may occur at system power up. The error number column is in the following format:

```
vvxxxx
mmm
```

Where "v" is a slot number, "x" is an error code number, and "m" is an ID number. The error codes for each ID code are listed below.

Table A-1 System Error Numbers

ID Number	Error Number	Description
(blank)	1?2???	Device not present and displayed
1	00	Keyboard is not functioning properly
	75	Keyboard has a stuck key
14	1	Unexpected interrupt or trap occurred with manual input device interface
	200	Time out occurred during loopback testing of manual input device interface
	201	Time out occurred while waiting for character to be received from manual input device interface
	202	Overrun or framing error with manual input device interface
	203	Data compare error with manual input device interface
17	1	Unexpected interrupt or trap occurred with printer interface
	200	Time out occurred during loopback testing of printer interface
	201	Time out occurred while waiting for character to be received from printer interface
	202	Overrun or framing error with printer interface
	203	Data compare error with printer interface

Table A-1 System Error Numbers (Cont)

ID Number	Error Number	Description
21	12	Time out occurred during data loopback testing of communication interface
	13	Unexpected interrupt or trap occurred during communication interface testing
	14	Unexpected external status interrupt from communication interface
	15	Special receive condition interrupt from communication interface
	16	Data compare error with communication interface
	17	Undefined interrupt from communication interface
	20	Time out occurred during modem loopback testing of communication interface testing
	21	Bad modem signals detected in communication interface
23	21	Clock did not interrupt in proper time
	24	Unexpected interrupt or trap occurred during clock testing
24	22	Data compare error with battery backed up RAM
	23	Unexpected interrupt or trap occurred during battery backed up RAM testing
25	1	No interrupts generated from interrupt controller
	2	Unexpected interrupt or trap occurred during clock interface testing
400	1	RD50 - Bad operation ended but on power up
	2	RD50 - Internal power-up self-test error
	3	RD50 - Bad bit/register sector, cylinder, or head registers
	4	RD50 - Error bit did not go away
	5	RD50 - Drive is not ready or seek incomplete
	6	RD50 - Restore command did not cause an "A" interrupt
	7	RD50 - Restore command did not set operation end bit
	10	RD50 - Error bit set Restore command
	11	RD50 - Incomplete read
	12	RD50 - Restore did not reach home during read test
	13	RD50 - Error bit set on operation end
	20	RD50 - Bad operation ended bit on power up
	21	RD50 - Seek incomplete, write fault, or drive not ready
	22	RD50 - Bad bit/register sector, cylinder, or head registers
	23	RD50 - Read command time out
	24	RD50 - Unexpected interrupt or trap
	25	RD50 - Data mark not found
	26	RD50 - Track 0 error
	27	RD50 - Illegal/aborted command
	30	RD50 - ID not found
	31	RD50 - CRC error, ID field
	32	RD50 - CRC error, data field

Table A-1 System Error Numbers (Cont)

ID Number	Error Number	Description
	33	RD50 - Unexpected operation end interrupt
	34	RD50 - Invalid operation end interrupt
	35	RD50 - Unexpected DRQ interrupt
	36	RD50 - Invalid DRQ interrupt
	37	RD50 - Restore command time out
500	1	RX50 - Internal self test time out
	2	RX50 - Unexpected interrupt or trap
	3	RX50 - Bad sector buffer
	10	RX50 - Bad drive 0 track 00 sensor
	20	RX50 - Bad drive 1 track 00 sensor
	30	RX50 - Both drives failed to respond
	40	RX50 - Tried to access an unspecified track number
	50	RX50 - Drive fails to see home
	60	RX50 - Data record not found
	70	RX50 - ID record not found
	100	RX50 - Time out for ID command done
	130	RX50 - Selected diskette is not ready
	140	RX50 - Diskette not installed correctly
	150	RX50 - ID CRC error
	170	RX50 - Seek error
	160	RX50 - Data ready signal (DRQ) did not respond in 27 ms
	120	RX50 - Bad ID read error
	200	RX50 - Data CRC error
	210	RX50 - Last data (DRQ) did not respond to DRQ within 27 ms
	220	RX50 - Tried to access an unavailable density
	230	RX50 - Drive not ready during write command
	240	RX50 - Drive not ready during read command
	250	RX50 - No sector matches the specified sector
	260	RX50 - Diskette write protected on a write command
	270	RX50 - Tried to access a wrong specified sector number
	300	RX50 - The lower nibble of RAM failed to pass memory test
	310	RX50 - The higher nibble of RAM failed to pass memory test
	320	RX50 - No index pulse detected
	330	RX50 - Drive speed not in limit
	340	RX50 - Bad format on a blank disk
	350	RX50 - Stepping error
	360	RX50 - Tried to set unsupported disk parameters
	370	RX50 - Phase Lock Loop (PLL) frequency not in limit
	400	RX50 - Tried to read a sector with a deleted data mark
	410	RX50 - Data buffer is bad
	420	RX50 - Tried to write a non-RX50 formatted disk

Table A-1 System Error Numbers (Cont)

HD Number	Error Number	Description
401	2	PC 100 video - Register failure
	3	PC 100 video - Plane 1 memory failure
	4	PC 100 video - Vertical retrace failure
	5	PC 100 video - Counter register failure
	6	PC 100 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 100 video - Plane 1 scroll register failure
	103	PC 100 I/O - Plane 2 memory failure
	106	PC 100 I/O - Plane 2 control failure
	107	PC 100 I/O - Plane 2 scroll register failure
	203	PC 100 I/O - Plane 3 memory failure
	206	PC 100 I/O - Plane 3 control failure
	207	PC 100 I/O - Plane 3 scroll register failure
1002	2	PC 350 video - Register failure
	3	PC 350 video - Plane 1 memory failure
	4	PC 350 video - Vertical retrace failure
	5	PC 350 video - Counter register failure
	6	PC 350 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 350 video - Plane 1 scroll register failure
	103	PC 350 I/O - Plane 2 memory failure
	106	PC 350 I/O - Plane 2 control failure
	107	PC 350 I/O - Plane 2 scroll register failure
	203	PC 350 I/O - Plane 3 memory failure
	206	PC 350 I/O - Plane 3 control failure
	207	PC 350 I/O - Plane 3 scroll register failure
1401	2	PC 350 video - Register failure
	3	PC 350 video - Plane 1 memory failure
	4	PC 350 video - Vertical retrace failure
	5	PC 350 video - Counter register failure
	6	PC 350 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 350 video - Plane 1 scroll register failure
	103	PC 350 I/O - Plane 2 memory failure
	106	PC 350 I/O - Plane 2 control failure
	107	PC 350 I/O - Plane 2 scroll register failure
	203	PC 350 I/O - Plane 3 memory failure
	206	PC 350 I/O - Plane 3 control failure
	207	PC 350 I/O - Plane 3 scroll register failure

Table A-1 System Error Numbers (Cont)

HD Number	Error Number	Description
10050	2	PC 100 video - Register failure
	3	PC 100 video - Plane 1 memory failure
	4	PC 100 video - Vertical retrace failure
	5	PC 100 video - Counter register failure
	6	PC 100 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 100 video - Plane 1 scroll register failure
	103	PC 100 I/O - Plane 2 memory failure
	106	PC 100 I/O - Plane 2 control failure
	107	PC 100 I/O - Plane 2 scroll register failure
	203	PC 100 I/O - Plane 3 memory failure
	206	PC 100 I/O - Plane 3 control failure
	207	PC 100 I/O - Plane 3 scroll register failure
20050	2	PC 350 video - Register failure
	3	PC 350 video - Plane 1 memory failure
	4	PC 350 video - Vertical retrace failure
	5	PC 350 video - Counter register failure
	6	PC 350 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 350 video - Plane 1 scroll register failure
	103	PC 350 I/O - Plane 2 memory failure
	106	PC 350 I/O - Plane 2 control failure
	107	PC 350 I/O - Plane 2 scroll register failure
	203	PC 350 I/O - Plane 3 memory failure
	206	PC 350 I/O - Plane 3 control failure
	207	PC 350 I/O - Plane 3 scroll register failure
10050	2	PC 350 video - Register failure
	3	PC 350 video - Plane 1 memory failure
	4	PC 350 video - Vertical retrace failure
	5	PC 350 video - Counter register failure
	6	PC 350 video - Plane 1 control, X, Y, or pattern register failure
	7	PC 350 video - Plane 1 scroll register failure
	103	PC 350 I/O - Plane 2 memory failure
	106	PC 350 I/O - Plane 2 control failure
	107	PC 350 I/O - Plane 2 scroll register failure
	203	PC 350 I/O - Plane 3 memory failure
	206	PC 350 I/O - Plane 3 control failure
	207	PC 350 I/O - Plane 3 scroll register failure
17777x	17x	Zero identification number occurred for longer than 20 seconds
17777x	17y	Non-existent memory trap occurred for longer than 20 seconds
Any HD	177777	Slot option detected (not displayed)
Any HD	17x	Slot option generated identification number, but slot option interface hardware indicates that option is not present
Any HD	17x	Slot option has a bad ROM

The following table defines the LED display during BSR execution in customer, console, or service mode. Please note that they are displayed in chronological order.

Table A-2 LED Display 1

LED	Definition
	System module
●	Memory daughterboard
	System module
●	EBO daughterboard
● ● ●	Slot 1
● ● ●	Slot 2
● ●	Slot 3
● ● ●	Slot 4
● ●	Slot 5
● ●	Slot 6
● ● ● ○	Power up self test complete

● means the light is off
○ means the light is on

The following table defines the light display during BSR execution in manufacturing mode. Please note that they are displayed in chronological order.

Table A-3 LED Display 2

LED	Definition
	JJ1
●	Onboard memory
●	Memory daughterboard
● ●	Interrupts
●	Clock and battery backed up RAM
● ●	Communications interface
● ●	Manual input device interface
● ● ●	Printer interface
● ●	Plane memory
● ●	Video gate array
● ● ●	Plane 2 memory
● ●	EBO 1 gate array
● ● ●	Plane 3 memory
● ● ●	EBO 2 gate array
●	Slots (excluding on board video)
● ● ● ●	Power up self-test complete

● means the light is off
○ means the light is on

The following table defines the light display after BSR execution in customer or console mode. Please note that they are displayed in a prioritized order where the most significant error takes precedence.

Table A-4 LED Display 3

LED	Definition
	System module
● ●	Unexpected interrupt or trap
●	Memory daughterboard
●	EBO daughterboard
● ● ●	Slot 1
● ● ●	Slot 2
● ●	Slot 3
● ● ●	Slot 4
● ●	Slot 5
● ●	Slot 6
● ●	Manual input device failure
● ●	No boot found
●	Video monitor not detected
●	Reserved
● ● ●	Reserved
● ● ● ●	No errors detected

● means the light is off
○ means the light is on

The following is a table of bugcheck codes defined for P/OS V2.0. A bugcheck occurs after the system is booted and is recognizable by the picture of the Professional computer system with nothing highlighted and two numbers (the Professional 380 has eight numbers) on the right side of the screen.

Table A-5 Bugcheck Codes

Code	Code Number	Definition
BF.PKS	000100/[illegible]	P/OS keyboard handler
BF.TTD	000200/[illegible]	Terminal driver
BF.PTS	1004[illegible]	P/OS terminal subsystem
BE.EXE	000300/[illegible]	Exec SSTSR general
BE.IOT	000300/000000	IOT in system state
BE.STK	000300/000001	Stack overflow
BE.BPT	000300/000002	Trace trap or breakpoint
BE.ILI	000300/000003	Illegal instruction trap
BE.ODD	000300/000004	Odd address or other trap 4
BE.SGF	000300/000005	Segment fault
BE.NPA	000300/000006	A task on P/OS without a parent (aborted)
BE.EMT	000300/000007	EMT trap
BE.TRP	000300/000010	TRAP trap
BF.UP	000400/[illegible]	System startup processing
BE.IN1	000400/000001	Can't install task CBOOT
BE.SP1	000400/000002	Can't spawn task CBOOT
BE.SP2	000400/000003	Can't spawn task CMAIN
BE.FNF	000400/000007	Required file not found

Professional/DECnet startup failure codes

BE.DSC	000400/000010	DSR corrupt
BE.BDP	000400/000011	Bad dispatch
BE.NWB	000400/000012	No way to boot via DECNA
BE.DAF	000400/000013	DSR allocation failure
BE.VIU	000400/000014	EMM vector in use
BE.NPD	000400/000015	Required PDV not found
BE.NSD	000400/000016	Required hardware not present

The following table is a list of the device IDs for the Professional 300 Series Computer System.

Table A-6 Device Identification Codes

ID	Type	Device Name
(000000)	(00)	Unready option
(000001)	(00)	Slot 0 (I/O backplane
(000002)	(00)	Slot 1 (I/O backplane
(000003)	(00)	Slot 2 (I/O backplane
(000004)	(00)	Slot 3 (I/O backplane
(000005)	(00)	Slot 4 (I/O backplane
(000006)	(00)	Slot 5 (I/O backplane
(000007)	(00)	Slot 6 (I/O backplane
(000010)	(00)	Slot 7 (I/O backplane
(000011)	(00)	CPU (80[illegible] processor (I/O slot 0)
(000012)	(00)	Floating point processor (FPP)
(000013)	(00)	CPU (82[illegible] FPP (integrated within CPU)
(000014)	(00)	VT100 keyboard control
(000015)	(00)	LK201 keyboard control
(000016)	(00)	VT100 keyboard
(000017)	(00)	LK201 keyboard
(000020)	(00)	Centronics printer port
(000021)	(00)	CTERM speaker control
(000022)	(00)	CTERM communications port
(000023)	(00)	Not applicable
(000024)	(00)	CTERM time-of-day clock
(000025)	(00)	CTERM nonvolatile RAM (NVR)
(000026)	(00)	CTERM interrupt controller
(000027)	(00)	CTERM DIAG/ROM version 1.0 (first release)
(000030)	(00)	CTERM DIAG/ROM version 2.0 (VMS)
(000031)	(00)	CTERM maintenance console port
(000032)	(00)	CTERM option present register
(000033)	(00)	CTERM serial number ROM
(000034)	(00)	CTERM monitor attachment
(000035)	(00)	CTERM primary RAM
(000036)	(00)	CTERM option RAM (256 kilobytes extended memory)
(000037)	(00)	80[illegible] base processor (I/O, NVR)
(000038)	(00)	80[illegible] interrupt controller (integrated inside I/O gate array)
(000039)	(00)	80[illegible] base system ROM version 1.0 (1st Release)
(000040)	(00)	CTERM DMA test module

Table A-6 Device Identification Codes (Cont)

ID	Type	Device Name
000041	000	CTI telephone management service (TMS)
000042	004	CTI Ethernet controller
000043	000	CTI /80/CPM softcard
000044	000	CTI T-11 softcard
000045	010	IVIS base module set
000046	000	HDDR HHH option (LDP)
000047	000	KANJI font module
000050	010	PC 152 bit map video base module
000051	000	DRC11-AA parallel interface (LDP)
000052	000	DLC11-AA serial interface (LDP)
000053	000	ARC11-AA analog interface (LDP)
000054	002	MRC11-AA ROM option (LDP)
000060	000	DH-6 touch module (DTM)
000401	002	CTI RD50 5 1/4" Winchester disk controller
001002	010	CTI bit map video base module
001003	000	CTI bit map video extension
002004	001	CTI RX50 5 1/4" floppy diskette controller
002405	004	CTI fast serial line
003006	000	IVIS system module (no ROM)
010001	000	Tempest keyboard (shielded)
010050	000	PC 152 bit map video extension
011002	000	IVIS bit map base module
011403	010	IVIS bit map video extension
030050	000	PC 152 bit map extension and color map
176775	000	Experimental ID (with identical bytes)
177176	000	Experimental ID (with identical bytes)
177775	000	Experimental ID
177776	000	Experimental ID
177777	000	Escape ID (will be used after all ID codes are exhausted)

The following table contains the I/O and IOSB codes

Table A-7 I/O and IOSB Codes

Signal	Decimal	Octal	Definition
IO.RAD	01	000001	Bad parameters
IO.IFC	02	000002	Invalid function code
IO.DNR	03	000003	Device not ready
IO.VER	04	000004	Parity error on device
IO.ONP	05	000005	Hardware option not present
IO.SPC	06	000006	Illegal user buffer
IO.DNA	07	000007	Device not attached
IO.DAA	08	000010	Device already attached
IO.DNC	09	000011	Device not attachable
IO.EOF	10	000012	End-of-file detected
IO.EOV	11	000013	End of volume detected
IO.WLK	12	000014	Write attempted to locked unit
IO.DAO	13	000015	Data overrun
IO.SRE	14	000016	Send/receive failure
IO.ABO	15	000017	Request terminated (see table at end)
IO.PRI	16	000020	Privilege violation
IO.RSU	17	000021	Sharable resource in use
IO.OVR	18	000022	Illegal overlay request
IO.BVI	19	000023	Odd byte count for virtual address
IO.BLK	20	000024	Logical block number too large
IO.MOD	21	000025	Invalid LUB module #
IO.CON	22	000026	LUB connect error
IO.NOD	23	000027	Caller's nodes exhausted
IO.DFU	24	000030	Device full
IO.IFU	25	000031	Index file full
IO.NSF	26	000032	No such file
IO.LCK	27	000033	Locked from read/write access
IO.HFU	28	000034	File header full
IO.WAC	29	000035	Accessed for write
IO.CKS	30	000036	File header checksum failure
IO.WAT	31	000037	Attribute control list I/O start error
IO.RER	32	000040	File processor device read error
IO.WER	33	000041	File processor device write error
IO.ALN	34	000042	File already accessed on LUN
IO.SNC	35	000043	File ID, file number check
IO.SQC	36	000044	File ID, sequence number check
IO.ALN	37	000045	Non-file accessed on LUN
IO.CLO	38	000046	File was not properly closed
IO.MBC	39	000047	(MFA) no buffer space available for file
IO.RBG	40	000050	Illegal record size

Table A-7 I/O and RMS Codes (Cont)

Signal	Decimal	Octal	Definition
IE.NRA	41	177727	File exceeds space allocated, no blocks
IE.IFC	42	177726	Illegal operation on fac descriptor block
IE.BTP	43	177725	Bad record type
IE.RAC	44	177724	Illegal record access bits set
IE.RAT	45	177723	Illegal record attributes bits set
IE.RCN	46	177722	Illegal record number - too large
IE.RCI	47	177721	Internal consistency error
IE.2DV	48	177720	Rename - 2 different devices
IE.FEX	49	177717	Rename - new file name already in use
IE.BDR	50	177716	Bad directory file
IE.RNM	51	177715	Can't rename old file system
IE.BDI	52	177714	Bad directory syntax
IE.FOP	53	177713	File already open
IE.BNM	54	177712	Bad file name
IE.BDV	55	177711	Bad device name
IE.BBK	56	177710	Bad block on device
IE.DUP	57	177707	ENTER - duplicate entry in directory
IE.STK	58	177706	Not enough stack space (HCS or F11)
IE.FHE	59	177705	Fatal hardware error on device
IE.NFI	60	177704	File ID was not specified
IE.SQO	61	177703	Illegal sequential operation
IE.EOT	62	177702	End of tape detected
IE.BVR	63	177701	Bad version number
IE.BHD	64	177700	Bad file header
IE.OFL	65	177677	Device off line
IE.BLC	66	177676	Block check, CRC, or framing error
IE.ONP	67	177675	Device online
IE.NSU	68	177674	No such node
IE.NLN	69	177673	Path lost to partner, this code must be odd
IE.DIS	69	177673	Path lost to partner, disconnected (Same as NLN)
IE.BLB	70	177672	Bad logical buffer
IE.TMM	71	177671	Too many outstanding messages
IE.NDR	72	177670	No dynamic space available, see also IE.UPN
IE.URJ	73	177667	Connection rejected by user
IE.NRJ	74	177666	Connection rejected by network
IE.EXP	-75	177665	File expiration date not reached
IE.BTF	-76	177664	Bad tape format
IE.NNC	77	177663	Not ANSI 'D' format byte count
IE.NDA	-78	177662	No data available
IE.NLK	-79	177661	Task not linked to specified RCS/RSX interrupts

Table A-7 I/O and RMS Codes (Cont)

Signal	Decimal	Octal	Definition
IE.NST	-80	177660	Specified task not installed
IE.UNS	-81	177657	Device offline, when still no request was issued
IE.IES	-82	177656	Invalid escape sequence
IE.PES	-83	177655	Partial escape sequence
IE.ALN	-84	177654	Alignment Error
IE.CLK	-85	177653	Clock error
IE.WCK	-86	177652	Write check failure
IE.NTR	-87	177651	Task not triggered
IE.MLT	-88	177650	Transfer terminated by receiving CPI
IE.ITC	-89	177647	Event flag already specified
IE.IRQ	-90	177646	I/O queue exceeded
IE.IOA	-91	177645	Inconsistent qualifier usage
IE.RIS	-92	177644	Circuit reset during operation
IE.TMF	-93	177643	Too many links to task
IE.NVL	-94	177642	Not a network UCB
IE.TKO	-95	177641	Timeout on request sent to TSK:
IE.CNR	-96	177640	Connection rejected
IE.UKN	-97	177637	Unknown error
IE.NFL	-98	177636	Unable to size device
IE.MDL	-99	177635	Media inserted incorrectly
IE.SPN	-100	177634	Spindown ignored
IS.PND	+00	0	Operation pending
IS.SUC	+01	1	Operation complete, success
IS.TNC	+02	2	Successful transfer but message truncated, receive buffer too small
IS.DAO	+03	3	Successful put with data overrun (not to be confused with IE.DAO)

I/O success codes
Low order byte of IS.SUC, high order byte is the termination character

IS.CR	+1139	002401	Carriage return was terminator
IS.ESC	+6911	015401	Escape (altmode) was terminator
IS.CC	+769	001401	Control-C was terminator
IS.ESQ	+59649	164401	Escape sequence was terminator
IS.PES	+32769	100001	Partial escape sequence terminator
IS.TOT	+1025	002001	TAB was terminator (block mode input)
IS.TAB	+2305	004401	Tab was terminator (terms mode input)
IS.TMO	+2	000002	Request timed out

Table A-7 I/O and IPSW Codes (4 out)

Signal	Decimal	Octal	Definition
IE.ABO related codes for mount/dismount failures			
IE.SYN	-07	000007	Syntax error
IE.NHM	-10	000012	Home block not found
IE.WRV	-11	000013	Wrong volume
IE.CHK	-12	000014	Checkpoint file still active
IE.SHA	-13	000015	Shadow recording still active
IE.MDM	-14	000016	Volume already marked for dismount
IE.MNT	-15	000017	Volume not mounted
IE.ALL	-16	000020	Volume already mounted FILES-11
IE.FOR	-17	000021	Volume already mounted foreign
Directive Error Codes			
IE.UPN	-01	177777	Insufficient dynamic storage, see also IE.NDR
IE.INS	-02	177776	Specified task not installed
IE.PTS	-03	177775	Partition too small for task
IE.UNS	-04	177774	Insufficient dynamic storage for send
IE.ULN	-05	177773	Unassigned LUN
IE.HWR	-06	177772	Device handler not resident
IE.ACT	-07	177771	Task not active
IE.ITS	-08	177770	Directive inconsistent with task state
IE.FIX	-09	177767	Task already fixed/unfixed
IE.CKP	-10	177766	Issuing task not checkpointable
IE.NCP	-11	177765	Task is checkpointable
IE.RBS	-15	177761	Receive buffer is too small
IE.PRI	-16	177760	Privilege violation
IE.RSU	-17	177757	Resource in use
IE.NSW	-18	177756	No swap space available
IE.ILV	-19	177755	Illegal vector specified
IE.ITN	-20	177754	Invalid table number
IE.LNF	-21	177753	Logical name not found
Codes -22 through -79 are reserved.			
IE.AST	-80	177600	Directive issued/not issued from AST
IE.MAP	-81	177657	Illegal mapping specified
IE.KIP	-83	177655	Window has I/O in progress
IE.ALG	-84	177654	Alignment error
IE.WOV	-85	177653	Address window allocation overflow

Table A-7 I/O and IPSW Codes (4 out)

Signal	Decimal	Octal	Definition
IE.NVR	-86	177652	Invalid region ID
IE.NVW	-87	177651	Invalid window ID
IE.ITP	-88	177650	Invalid IT parameter
IE.IBS	-89	177647	Invalid send buffer size (0:2048.)
IE.ULN	-90	177646	LUN locked in use
IE.IOP	-91	177645	Invalid IOPB
IE.IDU	-92	177644	Invalid device or unit
IE.ITI	-93	177643	Invalid time parameters
IE.PNS	-94	177642	Partition region not in system
IE.IPR	-95	177641	Invalid priority (0:250.)
IE.IUI	-96	177640	Invalid LUN
IE.IEF	-97	177637	Invalid event flag (0:64.)
IE.ADP	-98	177636	Part of IOPB out of user's space
IE.SDP	-99	177635	IOPB or IOPB size invalid
Success codes from directives placed in the directive status word			
ISCLR	0	0	Event flag was clear
ISSET	2	2	Event flag was set
ISSPD	2	2	Task was suspended
ISSUP	1	1	Logical name superseded

These are the errors from POSSE M. These codes are returned in word 1 of the status block upon return from a POSSE M call.

Table A-8 POSSE M Error Codes

Code	Code Number	Definition
+1	(octal)	Success
-1	177777	Directive error (word 2 contains SDSW)
-2	177776	I/O status error (word 2 = IOSB, word 3 = IOSB+2)
-3	177775	RMS error (word 2 contains STS, word 3 contains STV)
-4	177774	Server specific error, codes returned in words 2 through 7
-5	177773	Interface error, specific error in word 2

Interface error subcodes are returned in word 2 and are as follows:

-1	177777	Feature not supported
-2	177776	Impure area invalid, missing
-3	177775	Invalid number of parameters (too few, too many)
-4	177774	Server not installed (server name in words 2 and 3)
-5	177773	Illegal device specification
-6	177772	User buffer too small for returned data
-7	177771	POSSE M/task incompatibility error - relink task

NOTE

PRDATR and PRDDIR have no server specific error codes.

These are the server specific error codes from PROFIL:

+1	000001	Success
-1	177777	Illegal device
-2	177776	Device not in system
-3	177775	Failed to attach device
-4	177774	Block 0 bad - disk unusable
-5	177773	At least one of 3 BNs (0,24) is bad. Can't initialize
-6	177772	Bad block file overflow
-7	177771	Unrecoverable error
-8	177770	Device write locked
-9	177767	Device not ready
-10	177766	Failed to write bad block file
-11	177765	Privilege violation
-12	177764	Device is an alignment cartridge
-13	177763	Fatal hardware error
-14	177762	Allocation failure
-15	177761	I/O error sizing device
-16	177760	Allocation for sys file exceeds volume limit
-17	177757	Homeblock allocate write error
-18	177756	Bootblock write error - disk unusable
-19	177755	Index file bitmap I/O error

Table A-8 POSSE M Error Codes (Cont)

Code	Code Number	Definition
-20	177754	Bad block header I/O error
-21	177753	MFD file header I/O error
-22	177752	Null file header I/O error
-23	177751	Checkpoint file header I/O error
-24	177750	MFD write error
-25	177747	Storage bitmap file header I/O error
-26	177746	Failed to read bad block descriptor file
-27	177745	Volume name too long
-28	177744	Unrecognized disk type
-29	177743	Preallocation insufficient to fill first index file header
-30	177742	Preallocated too many headers for single header index file
-31	177741	Preallocation insufficient to fill 1st 2nd index file headers
-32	177740	Bad block limit exceeded for device
-33	177737	Device not resident
-34	177736	Bitmap too large - increase cluster factor
-35	177735	Storage bitmap I/O error
-36	177734	Homeblock I/O error
-37	177733	Index file header I/O error
-38	177732	Dismount of device failed
-39	177731	Cannot mount device foreign
-40	177730	Cannot mount device ODS-1
-41	177727	Cannot format RX/ - no software support
-42	177726	Cannot detach device
-43	177725	Checkpoint file header overflow, specify smaller checkpoint file
-44	177724	Illegal characters in volume name
-45	177723	Cannot format RX/ - no hardware support
-46	177722	Cannot format RX/ - speed out of range

These are the server specific error codes from PROTSK:

-1	177777	Error in parsing the SET TASK NAME string
-2	177776	Cannot determine type of service requested

These are the server specific error codes from PROTSK:

+1	000001	Successful install
-1	177777	Task name in use
-2	177776	File not found
-3	177775	Specified partition too small
-4	177774	Task and partition base mismatch
-5	177773	Length mismatch common block
-6	177772	Base mismatch common block
-7	177771	Too many common block requests
-8	177770	Checkpoint area too small
-9	177767	Not enough APRs for task image

Table 4-8 PRSSU Error Codes (Cont)

Code	Code Number	Definition
14	177762	Task not task image
15	177761	Base address must be on 4k boundary
16	177760	Illegal first APR
18	177756	Common block parameter mismatch
20	177754	Common block not loaded
22	177752	Task image virtual address overlaps common block
23	177751	Task image already installed
24	177750	Address extension not supported
26	177746	Checkpoint space too small, using checkpoint file
27	177745	No checkpoint space, assuming not checkpointable
29	177743	Illegal UIC
30	177742	No pool space
31	177741	Illegal use of partition or region
32	177740	Access to common block denied
33	177737	Task image I/O error
34	177736	Too many LUNs
35	177735	Illegal device
36	177734	Task may not be run
37	177733	Task active
39	177731	Task fixed
40	177730	Task being fixed
41	177727	Partition busy
43	177725	Common, task not in system
44	177724	Region or common fixed
45	177723	Can't do receive from requestor
46	177722	Can't attach to requestor
47	177721	Invalid request
48	177720	Can't return result parameter
49	177717	Error encountered on file open operation
50	177716	Error encountered on file close operation
51	177715	Can't get file LBN to process label blocks
52	177714	No taskname specified in batch request, no name in label block
53	177713	Unable to create or map to region

PRSSVC3 has the following server specific error codes

1	177777	File is not a system image
2	177776	Invalid boot device

The following table shows the VAX/VMS privileged error codes

Table 4-9 VAX/VMS Error Codes

Error Codes (octal)

Code Number	Definition
1	Dump is too big or loaded too high
2	Dump exited without saving state
3	Unexpected VAX trap
4	VAX trap referencing PCB
5	VAX trap referencing bitmap memory
6	Can't load Stop program in reserved memory
7	Program stopped by running it
10	Program didn't run or change the state
11	Bad stop memory
12	Program killed by memory test

Error codes defined by the base system module

174	Device not in system present register
175	IO read error
176	Error check failed on contents of ROM on device
177	Bus time-out trap reading device ID

CNOMBAO DMV11 NCTRL

....B1	....B5	....B9	....B13
....C1	....C5	....C9	....C13
....D1	....D5	....D9	....D13
....E1	....E5	....E9	....E13
....F1	....F5	....F9	....F13
....G1	....G5	....G9	....G13
....H1	....H5	....H9	....H13
....I1	....I5	....I9	....I13
....J1	....J5	....J9	....J13
....K1	....K5	....K9	....K13
....L1	....L5	....L9	....L13
....M1	....M5	....M9	....M13
....N1	....N5	....N9	....N13
....B2	....B6	....B10	....B14
....C2	....C6	....C10	....C14
....D2	....D6	....D10	....D14
....E2	....E6	....E10	....E14
....F2	....F6	....F10	....F14
....G2	....G6	....G10	....G14
....H2	....H6	....H10	....H14
....I2	....I6	....I10	....I14
....J2	....J6	....J10	....J14
....K2	....K6	....K10	....K14
....L2	....L6	....L10	....L14
....M2	....M6	....M10	....M14
....N2	....N6	....N10	....N14
....B3	....B7	....B11	....B15
....C3	....C7	....C11	
....D3	....D7	....D11	
....E3	....E7	....E11	
....F3	....F7	....F11	
....G3	....G7	....G11	
....H3	....H7	....H11	
....I3	....I7	....I11	
....J3	....J7	....J11	
....K3	....K7	....K11	
....L3	....L7	....L11	
....M3	....M7	....M11	
....N3	....N7	....N11	
....B4	....B8	....B12	
....C4	....C8	....C12	
....D4	....D8	....D12	
....E4	....E8	....E12	
....F4	....F8	....F12	
....G4	....G8	....G12	
....H4	....H8	....H12	
....I4	....I8	....I12	
....J4	....J8	....J12	
....K4	....K8	....K12	
....L4	....L8	....L12	
....M4	....M8	....M12	
....N4	....N8	....N12	