KMV11-C KMV11-C LOGIC DIAG
CVKMHA0

AH-T380A-MC
FICHE 1 OF 1

JAN 1984
digital
Made In USA

```
                .REM @
                                        IDENTIFICATION
                                        --------------

                PRODUCT CODE:   AC-T379A-MC

                PRODUCT NAME:   CVKMHA0 KMV11C LOGIC DIAG

                PRODUCT DATE:   AUG 1983

                MAINTAINER:     CSS ANNECY

                AUTHOR:         ALAIN FEUILLET

                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) 1983,1983 BY DIGITAL EQUIPMENT CORPORATION

                THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

                        DIGITAL         PDP             QBUS
                        DEC             DECUS           DECTAPE
```

```
                        /V01.01/        ALAIN FEUILLET
                        /V01.02/        NICOLE JACQUEMIN        16-FEB-83
                                        - DISPLAY CHECKSUM VALUE IN CASE OF ERROR
                                        - ADD ERROR MESSAGE WHEN MODE 2 CANNOE BE ENTERED
                                          FOR A STACK PROBLEM
                                        - DEFAULT VALUE FOR BAUD RATE IS 7
                                        - ADD TEST36 (AND RENAME LONG RAM TEST TO TEST37)
                                          WHICH TEST DATA RECEPTION USING THE POLLING
                                          FACILITY.
                                        - SUPPORT OF MORE THAN ONE UNIT.
                        /V01.03/        CHRISTIAN LEBRAUD       13-APR-83
                                        - CLEAN ERROR REPPORT FOR RELEASE
                                        - SUPPRESS INTERRUPT PRIORITY CHECKING
                        /V01.04/        CHRISTIAN LEBRAUD       27-MAY-83
                                        - CHANGE NXM TEST TO RUN ON A 11/23A
                                        - PROM P/N MUST BE DISPLAYED FOR EACH UNIT
                                          UNDER TEST
                                        - CLEAN PROBLEMS RELATED TO THE SUPERVISOR
                                          INTERFACE.
                        /V01.05/        CHRISTIAN LEBRAUD       1-JULY-83
                                        - CHANGE TEST33 TO RUN WITH 48KW
```

TABLE OF CONTENTS

TABLE OF CONTENTS

```
                        7.11 CONTROL OF THE KMV11-C EPROM PART NUMBER
                        7.12 SHORT RAM SPACE TEST
                        7.13 TEST OF THE INTERNAL KMV11 R/W REGISTERS
                        7.14 TEST INTERRUPT CAPABILITY OF KMV11 ON Q-BUS
                        7.15 TEST THAT Q-BUS ACCESS ON BSEL0/2 INTERRUPTS KMV
                        7.16 TEST OF THE REAL TIME CLOCK
                        7.17 TEST OF DATA TRANSMISSION - INTERRUPT MODE
                        7.18 TEST OF DATA RECEPTION - INTERRUPT MODE
                        7.19 TEST OF DATA RECEPTION - DUSTBIN
                        7.20 TEST OF DATA RECEPTION - ABORT CHARACTER
                        7.21 TEST OF DATA RECEPTION - CRC ERROR
                        7.22 TEST OF DATA RECEPTION - OVERRUN
                        7.23 TEST OF DMA IN TRANSFER USING SHORT FRAME
                        7.24 TEST OF DMA IN TRANSFER USING LONG FRAME
                        7.25 TEST OF DMA IN TRANSFER USING A NON EXISTENT MEMORY ADDRESS
                        7.26 TEST OF DMA OUT TRANSFER USING SHORT FRAME
                        7.27 TEST OF DMA OUT TRANSFER USING LONG FRAME
                        7.28 TEST OF DMA OUT TRANSFER USING A NON EXISTENT MEMORY ADDRESS
                        7.29 TEST DMA IN/OUT USING A SHORT FRAME WITH AN ODD BYTE COUNT
                        7.30 TEST DMA IN/OUT WITH LENGTH OF TRX AND RCV BUFFERS EQUAL
                        7.31 TEST DMA IN/OUT WITH LENGTH OF TRX AND RCV BUFFERS NOT EQUAL
                        7.32 TEST DMA IN/OUT USING A LONG FRAME
                        7.33 TEST DMA IN/OUT WITH A NON NULL EXTENDED ADDRESS IF HOST MEMORY LONG ENOUGH
                        7.34 TEST OF MODEM SIGNALS (EXTERNAL LOOP-BACK MODE)
                        7.35 EXECUTE A PROGRAM LOADED IN THE RAM
                        7.36 TEST OF DATA RECEPTION USING POLLING FACILITY
                        7.37 LONG RAM SPACE TEST
```

```
146
147
148
149
150
151                                 1.0 INTRODUCTION
152
153                                 1.1 PROGRAM ABSTRACT
154
155
156                                 THIS  DIAGNOSTIC IS  DESIGNED TO TEST OUT THE KMV11-C IN-
157                                 TERFACE WHICH IS PHYSICALLY A FULL MULTI-LAYER QUAD MODULE
158                                 NAMED M7502
159                                 THE MAJOR FEATURES OF THIS INTERFACE ARE:
160                                 - HIGH SPEED DATA COMMUNICATION INTERFACE FOR A Q-BUS
161                                 BASED SYSTEM.
162                                 - USE OF THE DCT11 MICROPROCESSOR.
163                                 - FIT INTO ANY POWERED BACKPLANE FOR Q-BUS SYSTEM WHICH
164                                 ACCEPTS QUAD MODULES.
165                                 - HIGH SPEED DATA TRANSFER BY USE OF A FIFO AND IN/OUT DMA
166                                 ACCESS COMPLETELY CONTROLLED BY HARDWARE.
167                                 - 8K BYTES OF EPROM CONTAINING EXECUTIVE AND ROOT FIRMWARE,
168                                 THE LATTER CONTAINING THE SELF TESTS AND TEST ROUTINES USED
169                                 WITH THIS DIAGNOSTIC.
170                                 - 32K BYTES OF RAM FOR THE LOADABLE APPLICATION FIRMWARE.
171                                 - A SET OF 16 CSR'S USED TO COMMUNICATE WITH THE HOST COM-
172                                 PUTER.
173                                 - INTERRUPT VECTORING.
174                                 - BIT STUFFING SYNCHRONOUS MODE OF TRANSMISSION.
175                                 - CONNECTION  TO A NETWORK USING THE RS-422 ELECTRICAL IN-
176                                 TERFACE.
177
178                                 THE PURPOSE OF  THIS PROGRAM IS TO PERFORM DIAGNOSTIC TES-
179                                 TING OF ALL THE  M7502 LOGIC IN A STATIC MANNER
180                                 THAT MEANS, TESTING OF:
181                                 - THE M7502 MEMORY  INCLUDING CSR'S,INTERNAL REGISTERS,THE
182                                 PROM AND RAM.
183                                 - Q-BUS LOGIC FOR DMA AND INTERRUPTS
184                                 - DCT11 INTERRUPT LOGIC
185                                 - LINE AND REAL TIME CLOCKS
186                                 - USART AND FIFO
187                                 - MODEM CONTROL AND MONITORING
188                                 - RS422 INTERFACE
189                                 A CONTROL IS  ALSO DONE TO VERIFY THAT THE PROGRAM AND THE
190                                 RESIDENT  FIRMWARE  HAVE THE  SAME REVISION. MOREOVER, THE
191                                 PROM PART NUMBER IS AUTOMATICALLY PRINTED DURING THE FIRST
192                                 PASS OF THE DIAGNOSTIC.
193                                 NOTICE THAT THE TEST ON MODEM SIGNALS REQUIRES THE  USE OF
194                                 AN EXTERNAL LOOP  BACK CONNECTOR AND IT WILL ONLY RUN  IF
195                                 THE OPERATOR HAS PREVIOUSLY  INSERTED THIS CONNECTOR.
196
197                                 IN ORDER TO TEST THE DIFFERENT LOGIC FUNCTIONS, TWO  MAIN-
198                                 TENANCE MODES ARE REQUIRED; MAINT MODE 2 AND 3.
199                                 - IN THE MAINTENANCE MODE 3, THE DCT11 IS INOPERATIVE  (IT
200                                 LOOPS ON ITSELF AND, LETS THE HOST TO WORK ON CSR'S ALONE)
201                                 - IN THE MAINTENANCE MODE 2, THE HOST ACTIVATES  SOME TEST
202                                 ROUTINES IN THE ROOT FIRMWARE AND LETS  THE DCT11 TO CHECK
```

THE HARDWARE FUNCTIONS BY ITSELF EXCEPT IN THE CASE OF DMA,
Q-BUS LOGIC AND CSR'S DATA TRANSFERS. THE MAINT MODE IS
SELECTED JUST AFTER THE MASTER CLEAR AND, FOR EACH TEST.

THE STATIC DIAGNOSTIC DOES NOT REQUIRE ANY MANUAL INTER-
VENTION DURING TEST PERIOD. A VISUAL CHECK OF THE TEST
PROCESSING IS POSSIBLE BY MEANS OF TWO MAINTENANCE LEDS
(GREEN AND YELLOW) ON THE KMV11-C DEVICE:
. GRN - APPLICATION FIRMWARE RUNNING
. YEL - SELFTEST RUNNING
. YEL+GRN - MAINTENANCE IS RUNNING
. GRN FLASH - APPLICATION ERROR
. YEL FLASH - SELF TEST ERRROR
. YEL+GRN FLASHING - MAINTENANCE ERROR
. BOTH LIGHT OFF - ERROR

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

THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM WILL ALLOW
MODIFICATION OF DEVICE PARAMETERS, SUCH AS Q-BUS ADDRESS,
VECTOR ADDRESS,LINE SPEED AND PROCESSOR TYPE.

1.2 HARDWARE INTRODUCTION

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE KMV11-C LOGIC
DIAGNOSTIC:

PDP11/23A,23B
24K MEMORY
CONSOLE TERMINAL
THE M7502 MODULE
ONE BC08S-10 CABLE WITH RS422 BERG/CINCH MODEM CONNECTION
BOX LOOP-BACK CONNECTORS(REMOTE AND LOCAL)

```
241
242
243
244
245
246                             3.0 PRELIMINARY PROGRAM REQUIREMENTS
247
248                             THE PROCESSOR AND MEMORY SHOULD BE THOROUGHLY  TESTED  PRIOR
249                             TO RUNNING THIS DIAGNOSTIC.
```

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 TOTAL TIME REQUIRED TO RUN THE M7502 STATIC TESTS IS LESS THAN 300 SECONDS FOR TESTS 1-35 AND ABOUT 1H15 FOR THE LONG RAM TEST, TEST36 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 NOT UTILIZED IN THIS PROGRAM. IF IT IS INSTALLED, IT IS DISABLED BY THE PROGRAM.

4.7 MEMORY PARITY OPTION

IF PARITY MEMORY IS INSTALLED, MEMORY PARITY TRAPS ARE DISABLED BY THE PROGRAM.

4.8 ERROR LOGGING

```
308                     THE  NUMBER  OF  ERRORS  WHICH  HAVE OCCURRED ON EACH DEVICE
309                     UNDER TEST SINCE THE LAST START OR RESTART COMMAND  IS  KEPT
310                     IN  AN  ERROR  LOG.   THIS  LOG  MAY BE PRINTED BY USING THE
311                     "PRINT" COMMAND (SEE SECTION 6.3.8).
```

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 PROMPT (DR>)
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:

```
200GDRS LOADED
DIAG.  RUN-TIME SERVICES REV. D APR-79
CVKMHAO KMV11-C LOGIC DIAGNOSTIC
UNIT IS M7502
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

```
443                          IDU   INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
444                          LOT   LOOP ON TEST
445
446                     THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO  0
447                     ARE CLEARED.  A FLAG NOT SPECIFIED IS CLEARED.  IF THE FLAGS
448                     SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED.  SEE  EXAMPLE  AT
449                     END OF 6.3.1.5.
450
451
452                     6.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)
453
454                     <INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF
455                     PASSES) IT IS DESIRED  THAT  THE  END  OF  PASS  MESSAGE  BE
456                     PRINTED.   THE  DEFAULT  IS  AT  THE END OF EVERY PASS.  SEE
457                     EXAMPLE AT END OF 6.3.1.5.
458
459
460                     6.3.1.5 EFFECT OF START COMMAND
461
462                     THE EFFECT OF THE START COMMAND IS TO INITIATE THE  HARDWARE
463                     PARAMETER  DIALOGUE,  THE  SOFTWARE  PARAMETER DIALOGUE, AND
464                     THEN THE DIAGNOSTIC TESTS THEMSELVES.
465
466                     THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE  QUESTION
467                     "#  UNITS?"  TO  WHICH  THE  OPERATOR REPLIES WITH A DECIMAL
468                     NUMBER N FROM 1 TO 16. THE TERM "UNIT" REFERS TO THE  DEVICE
469                     TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED.  FOLLOWING
470                     THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES  WILL
471                     BE  BUILT.  EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING
472                     ALL THE HARDWARE INFORMATION FOR  ONE  UNIT.   THE  OPERATOR
473                     MUST  SUPPLY  N  (NUMBER OF UNITS) VALUES FOR EACH QUESTION.
474                     HE MAY DO THIS BY GIVING ONE ANSWER  TO  EACH  QUESTION  (IN
475                     WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR
476                     BY GIVING N VALUES, SEPARATED BY COMMAS,  TO  EACH  QUESTION
477                     (SERIES  WILL  BE POSED ONCE).  EACH QUESTION IS FOLLOWED BY
478                     THE RESPONSE RADIX (D FOR  DECIMAL,  B  FOR  BINARY,  O  FOR
479                     OCTAL,  L  FOR  YES/NO) IN PARENTHESES AND THE DEFAULT VALUE
480                     AFTER THE PARENTHESES.
481                     FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE  QUESTIONS
482                     TO  BUILD  THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK
483                     VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.
484
485                     WHEN THE QUESTION "# UNITS?" IS ANSWERED, MEMORY STORAGE  IS
486                     ALLOCATED  FOR  THE  P-TABLES, AND IF THERE IS NOT ENOUGH TO
487                     ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED.  IN
488                     THIS  CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO
489                     TEST ALL UNITS.
490
491                     EXAMPLE:
492
493                     STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE
494
495                     THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE,  EACH  PASS
496                     CONSISTING  OF  TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST
497                     ALL UNITS.  THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND
498                     SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON
499                     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

```
557                     THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART.
558                     IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.
559
560
561                     6.3.3.2 FLAG SWITCH (/FLAGS:<FLAG-LIST>)
562
563                     <FLAG-LIST>  IS  SAME  AS  IN START COMMAND, BUT UNSPECIFIED
564                     FLAGS RETAIN THEIR CURRENT VALUE.
565
566
567                     6.3.3.3 EFFECT OF CONTINUE COMMAND
568
569                     CONTINUE MUST FOLLOW A START OR RESTART,  AND  COMMAND  MODE
570                     MUST  HAVE  BEEN  ENTERED  DUE  TO  A  HALT  ON  ERROR  OR A
571                     CONTROL/C. THE EFFECT  OF  THE  COMMAND  IS  TO  GO  TO  THE
572                     BEGINNING  OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT
573                     OR CONTROL/C TOOK PLACE.  SOFTWARE DIALOGUE  MAY  OPTIONALLY
574                     BE REEXECUTED.  HARDWARE PARAMETERS MAY NOT BE CHANGED.
575
576
577                     6.3.4 PROCEED COMMAND
578
579                     ************************************************************
580                     PRO(CEED)/FLAGS:<FLAG-LIST>
581                     ************************************************************
582
583
584                     6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
585
586                     <FLAG-LIST>  IS  AS  IN  THE  START COMMAND, BUT UNSPECIFIED
587                     FLAGS RETAIN THEIR CURRENT VALUE.
588
589
590                     6.3.4.2 EFFECT OF PROCEED COMMAND
591
592                     PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE.   COMMAND
593                     MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR.  THE EFFECT
594                     OF  THE  COMMAND  IS  TO  BEGIN  EXECUTION  AT  THE  LOCATION
595                     FOLLOWING  THE  ERROR  CALL.   NEITHER HARDWARE NOR SOFTWARE
596                     PARAMETERS MAY BE ALTERED.
597
598
599                     6.3.5 ADD COMMAND
600
601                     ************************************************************
602                     ADD/UNITS:<UNIT-LIST>
603                     ************************************************************
604
605
606                     6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>
607
608                     <UNIT-LIST> IS AS IN THE RESTART COMMAND.
609
610
611                     6.3.5.2 EFFECT OF ADD COMMAND
612
613                     THE UNITS SPECIFIED ARE ADDED TO THE  TEST  SEQUENCE.   EACH
```

```
614                     UNIT  MUST  HAVE  A  P-TABLE  IN  MEMORY  DUE  TO AN EARLIER
615                     HARDWARE DIALOGUE.  THIS  COMMAND  MUST  BE  FOLLOWED  BY  A
616                     RESTART  OR  CONTINUE.   THE UNITS SWITCH MUST BE SPECIFIED.
617                     THE ADD COMMAND IS  MEANINGFUL  ONLY  FOR  UNITS  THAT  WERE
618                     PREVIOUSLY DROPPED.
619
620
621                     6.3.6 DROP COMMAND
622
623                     ************************************************************
624                     DRO(P)/UNITS:<UNIT-LIST>
625                     ************************************************************
626
627
628                     6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
629
630                     <UNIT-LIST> IS AS IN THE RESTART COMMAND.
631
632
633                     6.3.6.2 EFFECT OF DROP COMMAND
634
635                     THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING.  THE UNITS
636                     WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR  START
637                     COMMAND.   THE  UNITS  SWITCH MUST BE ENTERED.  THIS COMMAND
638                     MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.
639
640
641                     6.3.7 PRINT COMMAND
642
643                     ************************************************************
644                     PRI(NT)
645                     ************************************************************
646
647
648                     6.3.7.1 EFFECT OF PRINT COMMAND
649
650                     THE TOTAL NUMBER OF ERRORS FOR  EACH  UNIT  SINCE  THE  LAST
651                     START  OR  RESTART  COMMAND  ARE  PRINTED.  THE ISR (INHIBIT
652                     STATISTICAL REPORTING) FLAG IS CLEARED.
653
654
655                     6.3.8 DISPLAY COMMAND
656
657                     ************************************************************
658                     DIS(PLAY)/UNITS:<UNIT-LIST>
659                     ************************************************************
660
661
662                     6.3.8.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
663
664                     <UNIT-LIST> IS AS IN THE RESTART COMMAND.
665
666
667                     6.3.8.2 EFFECT OF DISPLAY COMMAND
668
669                     THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST  ARE  PRINTED
670                     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- INITIAL DIALOGUE (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 4 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. KMV11-C CSR ADDRESS : (O) 177000 ?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SELO) RESIDE ON THE QBUS. THE ALLOWABLE RANGE IS 160000-177776

(OCTAL), AND THE DEFAULT IS 177000.

2. DEVICE VECTOR ADDRESS : (O) 300 ?

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

3. LOOPBACK CONNECTOR USED : (L) Y ?

THIS TELLS THE PROGRAM IF THE LOOP-BACK CONNECTOR IS PLUGGED
OR NOT. IF IT IS NOT THE INTERNAL LOOPBACK IS AUTOMATICALLY SELECTED
THE ALLOWABLE VALUES ARE Y OR NO, AND THE DEFAULT IS YES.

4. DEVICE LINE CLOCK RATE SELECTED : (O) 7 ?

THIS TELLS THE PROGRAM AT WHICH CLOCK RATE THE TRANSMISSION
LINE WILL BE TESTED (K BIT PER SECONDS). THE ALLOWABLE VALUES ARE:

        0=2,4K
        1=9,6K
        2=19.2K
        3=48K
        4=56K
        5=64K
        6=72K
        7=ALL CLOCK RATES

THE DEFAULT VALUE IS 7.

5. DO YOU WANT TO RUN LONG RAM TEST : (L) N ?

THIS TELLS THE PROGRAM IF THE LONG RAM TEST HAS TO BE RUN.
IF ANSWER IS YES TEST DURATION IS 1H15.

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETERS ARE REQUESTED BY THE LOGIC DIAGNOSTIC

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.

AFTER THE INITIAL DIALOGUE, THE SUPERVISOR ISSUES THE QUES-
TION BELOW:

CHANGE HW (L)   ?

IF YOUR RESPONSE IS Y (YES), THE SUPERVISOR ASKS FOR THE
NUMBER OF UNITS UNDER TEST:

UNITS (D) ?

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
                NAMED VALUE.

                A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR
                EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS
                SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN
                INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE
                RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).
                NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO
                CONSTRUCT A SET OF P-TABLES.  ASSUME THAT WE HAVE 16 UNITS,
                AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE
                SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS IN THE
                DIALOGUE).  LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE
                THE NUMBER 75 FOR ALL 16 TABLES.  LET THE DESIRED VALUE FOR
                THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER
                (0,1,2,...,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE THE
                VALUE 11.  LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE
                THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE NUMBER 77 FOR
                THE LAST 9 UNITS.

                THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

                # UNITS (D) ? 16

                UNIT 1
                <QUESTION 1> ? 75
                <QUESTION 2> ? 0-6
                <QUESTION 3> ? 76

                UNIT 21
                <QUESTION 1> ?
                <QUESTION 2> ? 7-11,,13-15
                <QUESTION 3> ? 77

                THE FIRST TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75
```

```
                        IN ALL 16 TABLES.  SLOT TWO RECEIVES THE VALUES  0,1,2,....,6
                        IN  TABLES  0  THRU  6 AND A CONSTANT 6 IN TABLES 7 THRU 15.
                        SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.

                        THE SECOND TIME THRU THE SERIES, TABLES 16 THRU THE END  ARE
                        GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS
                        PRINTED OUT FOR THE THE OPERATOR IN THE FORM  "UNIT  XX"  AT
                        THE  BEGINNING  OF EACH SERIES).  QUESTION 1 IS RESPONDED TO
                        BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 7 THRU
                        15, SINCE NO NEW EXLICIT VALUES ARE TYPED IN.  SLOT TWO GETS
                        THE VALUES 7,8,9,10,11 IN TABLES 7 THRU 11, AND GETS A 11 IN
                        SLOT  12, AND GETS THE VALUFS 13,14,15 IN TABLES 13 THRU 15.
                        SLOT THREE GETS THE VALUE 77 IN TABLES 7 THRU 15.

                        THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT
                        16 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION
                        (NAMELY QUESTION 2).
```

7.0 TEST DESCRIPTIONS

7.1 TEST01

** - TEST Q-BUS ACCESS ON ALL KMV11 CSR'S

DESCRIPTION: VERIFY THAT REFERENCING KMV11 CSR'S DOES NOT CAUSE A TIME OUT TRAP

REPORT: ERROR 1 TIME-OUT ERROR

7.2 TEST02

** - TEST THAT ALL CSR'S CAN BE CLEARED

DESCRIPTION: THE HOST CLEARS ALL CSR'S THEN IT TESTS THEM FOR ALL ZERO.

MAINT MODE: 3

REPORTS: ERROR 2 MASTER CLEAR FAILS TO RESET
ERROR 3 ALL CSR'S CAN'T BE CLEARED

7.3 TEST03

** - TEST THAT THE KMV11 CAN BE INITIALIZED

DESCRIPTION: THE HOST SETS SEL0 TO ENTER MAINTENANCE MODE3 AFTER MASTER CLEAR.
UPON COMPLETION OF INITIALISATION, THE DCT11 MUST CLEAR MCLR THEN ENTER MODE 3(THE DCT11 LOOPS ON ITSELF)

REPORT: ERROR 2 MASTER CLEAR FAILS TO RESET

7.4 TEST04

** - TEST Q-BUS ACCESS ON SEL2 TILL SEL36

DESCRIPTION: THE HOST WRITES PATTERNS IN EACH CSR THEN IT CHECKS THE SELECTED CSR FOR THE PATTERN AND ALL OTHERS CSR'S FOR ZERO. BEFORE SELECTION OF A NEW CSR FOR TEST, ALL CSR'S ARE CLEARED

```
                        MAINT MODE:     3

                        PATTERNS:       ALL 0
                                        ALL 1
                                        052525
                                        125252
                                        ROTATING 1
                                        ROTATING 0

                        REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 4         DATA COMPARE ERROR ON CSR'S
                                                        Q-BUS ACCESS



                    7.5  TEST05

                     ** - TEST Q-BUS ACCESS ON SELO CSR

                     DESCRIPTION:   THE  HOST CLEARS  ALL CSR'S THEN WRITES ONE
                                    PATTERN IN SELO SUCH AS:
                                            SELO = PATTERN & 1'S  COMP  OF MCLR
                                    THEN THE HOST CHECKS SELO FOR PATTERN & 1'S
                                    COMP MCLR AND ALL OTHER CSR FOR ZERO.

                     MAINT MODE:    3

                     PATTERNS:      ALL 0
                                    ALL 1
                                    052525
                                    125252
                                    ROTATING 1
                                    ROTATING 0

                     REPORTS:       ERROR 2         MASTER FAILS TO RESET
                                    ERROR 5         DATA COMPARE ERROR ON SELO



                    7.6  TEST06

                     ** - TEST DATA TRANSFER THROUGH SELO

                     DESCRIPTION:   TEST IN TWO PARTS:
                                                    THE FIRST TRIES TO ENTER
                                    CORRECTLY MAINTENANCE MODE 2(MAINT1!) AFTER
                                    MASTER CLEAR, GIVEN THE DCT11 THE POSSIBILI
                                    TY TO RUN TEST ROUTINES.THE HOST SETS  SELO
                                    WITH MAINTENANCE MODE 2 AND THE PATTERN 252
                                    IN THE LOW BYTE.THE DCT11 MUST READ IT THEN
                                    WRITE BACK SELO WITH MCLR  CLEARED AND  125
                                    IN THE LOW BYTE.  THE SECOND TESTS THAT THE
                                    DCT11 CAN READ THE  RUN,READ,WRITE BITS AND
                                    PATTERN 125 IN THE LOW BYTE. THE DCT11 MUST
                                    CLEAR ALL BITS AND WRITE BACK IN SELO ERROR
```

```
                                        BIT AND PATTERN 252.

                                MAINT MODE:     2

                                KMV ROUTINES:   25 AND 52

                                REPORTS:        ERROR 6         DATA COMPARE ERROR ON SELO,
                                                                THE DCT11 CAN'T CRRECTLY AC
                                                                CESS SELO OR ENTER MAINTANANCE
                                                                MODE 1.



                        7.7  TEST07

                         ** - TEST THAT ALL THE NATIVE FIRMWARE CAN BE ACCESSED

                         DESCRIPTION:   THE HOST ASKS THE DCT11  TO  CALCULATE  AND
                                        VERIVYING CHECKSUM  OF THE NATIVE CODE. THE
                                        HOST WAITS FOR 500MS AND POLLS SELO FOR THE
                                        DCT11 ANSWER:
                                                SELO=TEST NUMBER IF NO KMV11 ANSWER
                                                SELO=0 IF TEST OKE
                                                SELO=100 IF ANY ERROR

                         MAINT MODE:    2

                         KMV ROUTINE:   01

                         REPORTS:       ERROR 11        NO  ANSWER  FROM  THE KMV11
                                        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 7         CHECKSUM ERROR



                        7.8  TEST08

                         ** - TEST THE NATIVE FIRMWARE REVISION CODE

                         DESCRIPTION:   THE HOST ASKS THE DCT11 FOR THE CURRENT REV
                                        ISION CODE.
                                        THE DCT11 USES SELO TO ANSWER:
                                                SELO=TEST NUMBER IF NO KMV11 ANSWER
                                                SELO=1 TO 63 FOR REVISION
                                        AND THE HOST COMPARES IT AGAINST THE ACTUAL
                                        ONE AT LOCATION REVCOD.

                         MAINT MODE:    2

                         KMV ROUTINE:   23,24

                         REPORTS:       ERROR 11        KMV11 CAN'T ANSWER
                                        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 10        BAD REVISION CODE
```

7.9 TEST09

** - TEST DATA TRANSFER THROUGH REGISTER SEL2

DESCRIPTION: THE HOST WRITES ONE PATTERN IN SEL2, WHICH MUST BE READ BY THE DCT11.THE DCT11 ANSWERS BY LOADING IN SEL0 THE PATTERN CODE CORRESPONDING TO THE PATTERN IT HAS READ.

	PATTERN CODE FOR	PATTERN
SEL0 =	0	ALL 0
	1	ALL 1
	2	052525
	3	125252
	4 - 19.	ROTATING 0
	20. - 35.	ROTATING 1
	36.	UNKNOWN

THEN THE HOST TESTS THE CODE RETURNED.

MAINT MODE: 2

KMV ROUTINE: 03

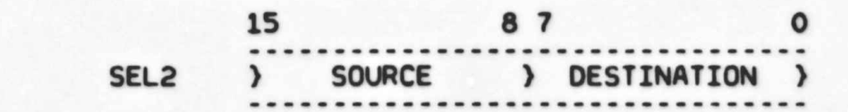
REPORTS:

ERROR 11	NO ANSWER FROM THE KMV11
ERROR 2	MASTER CLEAR FAILS TO RESET
ERROR 12	DCT11 CAN'T CORRECTLY READ SEL2

7.10 TEST10

** - TEST DATA TRANSFER THROUGH ALL CSR'S EXCEPT SEL0

DESCRIPTION: THE HOST WRITES ONE PATTERN IN ONE CSR AND ASKS THE DCT11 TO READ THEN TO WRITE IT IN A SECOND CSR.THE DCT11 INDICATES COMPLETION OF THE TRANSFER BY CLEARING SEL0. THEN THE HOST TESTS THE SELECTED CSR'S FOR PATTERN AND ALL OTHER CSR'S FOR ZERO.
ALL CSR ARE CLEARED BEFORE SELECTION OF NEW CSR'S FOR TEST.
THE HOST USES SEL2 TO PASS CSR OFFSET OF A SOURCE AND DESTINATION, SUCH AS:



THE TEST INCLUDES THE FOLLOWING TRANSFERS:

FROM	TO

```
                                                        SEL4            SEL2
                                                        SEL2            SEL4
                                                        SEL4            SEL6
                                                        SEL6            SEL10
                                                        SEL10           SEL12
                                                        SEL12           SEL14
                                                        SEL14           SEL16
                                                        SEL16           SEL20
                                                        SEL20           SEL22
                                                        SEL22           SEL24
                                                        SEL24           SEL26
                                                        SEL26           SEL30
                                                        SEL30           SEL32
                                                        SEL32           SEL34
                                                        SEL34           SEL36
                                                        SEL36           SEL4

                        MAINT MODE:     2

                        KMV ROUTINE:    04

                        REPORTS:        ERROR 2         MASTER  CLEAR FAILS TO RESET
                                        ERROR 11        KMV11 CAN'T ANSWER
                                        ERROR 13        DCT11 CAN'T CORRECTLY ACCESS
                                                        CSR'S


7.11 TEST11

 ** - CONTROL OF THE KMV11-C EPROM'S PART NUMBER

 DESCRIPTION:   THIS CONTROL IS TAKEN OUT ON THE  FIRST PASS
                ONLY.THE GOAL IS TO GET THE 2 PART NUMBERS STO-
                RED IN THE KMV EPROM THEN TO PRINT IT.
                IN ORDER TO DO THAT, THE HOST ASKS THE DCT11
                TO RETURN BACK THE PART NUMBER ADDRESS, SUCH
                AS:
                        SEL0 = 0 FOR KMV ROUTINE 27 JOB DONE
                        SEL2 = PART NUMBER ADDRESS

                THEN THE HOST BY MEANS OF THE  KMV 'MAILBOX'
                ROUTINE READS ALL PART NUMBER CHARACTERS. FOR
                THIS ROUTINE THE HOST PASSES PARAMETERS:

                        SEL2 = SEL10 ADDRESS
                        SEL4 = PART NUMBER ADDRESS
                        SEL6 = 4 FOR READ BYTE INDICATION
                        SEL10= WILL BE  LOADED BY  THE DCT11
                               WITH A CHARACTER
                THEN THE HOST PRINTS THE LOW AND HIGH PROM NUMBERS.

 MAINT MODE:    2

 KMV ROUTINES:  23,24
```

```
                        REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 11        NO KMV11 RESPONSE



                    7.12 TEST12

                     ** - SHORT RAM SPACE TEST

                     DESCRIPTION:   THE HOST ASKS THE  DCT11 TO  TEST  THE  RAM
                                    BY ITSELF. PERIODICALLY THE HOST CHECKS THE
                                    REGISTER SEL0 FOR ANY DCT11 RESPONSE:
                                            SEL0 = 0        IF TEST SUCCEEDED
                                            SEL0 = 100      IF ANY ERROR
                                    IN THE LATTER CASE, THE FOLLOWING REGISTERS
                                    CONTAINS THE ERROR REPORTS,SUCH AS:

                                    - IF THE DCT11 CAN'T CLEAR ALL THE RAM
                                            SEL2 = 2
                                            SEL4 = FIRST BAD WORD ADDRESS
                                            SEL6 = FIRST BAD WORD

                                    - IF DATA COMPARE ERROR AT LOCATION TESTED
                                            SEL2 = 4
                                            SEL4 = RAM ADDRESS
                                            SEL6 = BAD WORD
                                            SEL10= PATTERN

                     MAINT MODE:    2

                     KMV ROUTINE:   26

                     REPORTS:       ERROR 2         MASTER CLEAR FAILS TO RESET
                                    ERROR 11        NO ANSWER FROM THE KMV11
                                    ERROR 14        ALL THE RAM CAN'T BE CLEARED
                                    ERROR 15        DATA COMPARE ERROR ON RAM AT
                                                    LOCATION UNDER TEST


                    7.13 TEST13

                     ** - TEST OF INTERNAL KMV11 R/W REGISTERS

                     DESCRIPTION:   THE HOST ASKS THE  DCT11 TO  TEST  ALL  R/W
                                    REGISTERS BY ITSELF. PERIODICALLY, THE HOST
                                    POLLS SEL0 FOR ANY DCT11 RESPONSE:

                                            SEL0 = 0        FOR TEST OKE
                                            SEL0 = 100      FOR ANY ERROR

                                    IN THE LATTER CASE, THE FOLLOWING REGISTERS
                                    CONTAINS ERROR REPORTS, SUCH AS:

                                            SEL2 = 2        BYTE COUNT OUT REG.
                                                 = 4        BUS ADDRESS OUT REG.
```

```
1202                                                    = 6         EXT ADDRESS OUT REG.
1203                                                    = 10        BYTE COUNT IN REG.
1204                                                    = 12        BUS ADDRESS IN REG.
1205                                                    = 14        EXT ADDRESS IN REG.
1206                                                SEL4 = CURRENT PATTERN
1207                                                SEL6 = BAD WORD
1208
1209                                    MAINT MODE:     2
1210
1211                                    KMV ROUTINE:    06
1212
1213                                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1214                                                    ERROR 11        NO  RESPONSE FROM THE KMV11
1215                                                    ERROR 17        DATA COMPARE ERROR ON KMV11
1216                                                                    INTERNAL R/W REGISTER
1217
1218
1219
1220
1221                                    7.14 TEST14
1222
1223                                    ** - TEST INTERRUPT CAPABILITY OF KMV11-C ON Q-BUS
1224
1225                                    DESCRIPTION:    THE  HOST  GETS  ACTUAL VECTOR AND PRIORITY
1226                                                    FROM P.TABLE. THE HOST PREPARES VECTOR AREA
1227                                                    AND SETS ITS PRIORITY TO 7 THEN IT ASKS THE
1228                                                    DCT11  TO  REQUEST  AN  INTERRUPT ON Q-BUS.
1229                                                    SINCE THE DCT11 CAN  REQUEST  INTERRUPT  ON
1230                                                    Q-BUS THROUGH  VECTORS XX0 AND XX4 THE TEST
1231                                                    IS COMPOSED  OF TWO SUBTESTS, ONE  FOR EACH
1232                                                    VECTOR. THE HOST INDICATES TO THE DCT11 THE
1233                                                    VECTOR UNDER TEST BY MEANS OF SEL2:
1234                                                            SEL2 = 0        FOR XX0
1235                                                            SEL2 = 1        FOR XX4
1236                                                    THE DCT11 RESPONSE AFTER COMPLETION IS:
1237                                                            SEL0 = 0
1238                                                    THEN,
1239                                                    FOR LEVEL=6 TO 3 DO
1240                                                            NOP
1241                                                            NOP
1242                                                            READ INTERRUPT FLAG
1243                                                            IF FLAG SET THEN
1244                                                                    IF AT LEGAL PRIORITY THEN
1245                                                                            GOOD EXIT
1246                                                                    ELSE
1247                                                                            ERROR REPORTS
1248                                                                    ENDIF
1249                                                            ENDIF
1250                                                    ENDDO
1251                                                    REPORT ERROR FOR NO INTERRUPT
1252
1253                                    MAINT MODE:     2
1254
1255                                    KMV ROUTINE:    07
1256
1257                                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1258                                                    ERROR 11        NO KMV11 RESPONSE
```

```
1259                                            ERROR 20        Q-BUS INPERRUPT AT AN UNEXPECTED PRIORITY LEVEL
1260                                            ERROR 21        NO Q-BUS INTERRUPT OCCURS
1261
1262
1263
1264
1265                                    7.15 TEST15
1266
1267                                     ** - TEST THAT ACCESS TO BSEL0/BSEL2 INTERRUPTS THE DCT11
1268
1269                                     DESCRIPTION:   THE HOST ASKS THE  DCT11 TO  SERVICE  INTER-
1270                                                    RUPT  ON  BSEL0/BSEL2 INTERRUPT. SINCE BSEL0
1271                                                    AND BSEL2 ACCESSES CAN PRODUCE AN INTERRUPT,
1272                                                    THE HOST USES  SEL2  TO SPECIFY TO THE DCT11
1273                                                    WHICH CSR ACCESS IS TESTED:
1274                                                            SEL2 = 0        FOR BSEL0 ACCESS
1275                                                            SEL2 = 2        FOR BSEL2 ACCESS
1276                                                    DCT11 RESPONSE WILL BE:
1277                                                            SEL0 = 0     IF EXPECTED INTERRUPT
1278                                                                 = 100   IF UNEXPECTED INTERRUPT
1279                                                                 = OTHER IF NOT INTERRUPTED
1280                                                    THEN AFTER ASKING THE DCT11,
1281                                                    FOR BSEL=0 TO 37 EXCEPT BSEL0/2 DO
1282                                                            SET BITS IN BSEL
1283                                                            WAIT FOR 1 MS
1284                                                            READ SEL0
1285                                                            CASE SEL0 OF
1286                                                                    0 : REPORT ERROR 23 AND EXIT
1287                                                                  100 : REPORT ERROR 23 AND EXIT
1288                                                            ENDCASE
1289                                                    ENDDO
1290                                                    SET BITS IN BSEL0/2
1291                                                    WAIT FOR 1 MS
1292                                                    READ SEL0
1293                                                    IF SEL0=0 THEN
1294                                                            GOOD EXIT
1295                                                    ELSE
1296                                                            CASE SEL2 OF
1297                                                                    2 : REPORT ERROR 24 AND EXIT
1298                                                                    4 : REPORT ERROR 24 AND EXIT
1299                                                                OTHER : REPORT ERROR 22 AND EXIT
1300                                                            ENDCASE
1301                                                    ENDIF
1302
1303                                     MAINT MODE:    2
1304
1305                                     KMV ROUTINE:   10
1306
1307                                     REPORTS:       ERROR 2         MASTERCLEAR FAILS TO RESET
1308                                                    ERROR 11        NO KMV11 RESPONSE
1309                                                    ERROR 22        DCT11 NOT INTERRUPTED
1310                                                    ERROR 23        NOT EXPECTED CSR CAUSES AN
1311                                                                    INTERRUPT ON DCT11
1312                                                    ERROR 24        KMV11 INTERRUPTED AT ILLEGAL VECTOR
1313
1314
1315
```

```
1316
1317                                    7.16 TEST16
1318
1319                                    ** - TEST OF THE REAL TIME CLOCK
1320
1321                                    DESCRIPTION:    THE  HOST  ASKS THE  DCT11 TO  TEST  THE REAL
1322                                                    TIME CLOCK BY ITSELF. THE  DCT11 TEST
1323                                                    THE REAL TIME CLOCK INTERRUPT
1324                                                    FOR TWO PERIODS OF CLOCK THEN IT DISABLES IT
1325                                                    PERIODICALLY THE HOST POLLS SEL0 FOR ANY
1326                                                    DCT11 RESPONSE:
1327
1328                                                            SEL0 = 0        IF TEST OKE
1329                                                            SEL0 = 100      IF ANY ERROR
1330
1331                                                    IN THE LATTER CASE, THE FOLLOWING CSR'S CON-
1332                                                    TAIN THE ERROR REPORT SUCH AS:
1333
1334                                                    - DCT11 INTERRUPTED BEFORE TIME ELAPSED
1335                                                            SEL2 = 2
1336                                                            SEL4 = CLOCK PERIOD
1337
1338                                                    - COUNTER DECREMENTED BUT NO INTERRUPT
1339                                                            SEL2 = 4
1340                                                            SEL4 = CLOCK PERIOD
1341
1342                                                    - COUNTER STOPPED
1343                                                            SEL2 = 6
1344                                                            SEL4 = CLOCK PERIOD
1345
1346                                                    - NOT EXPECTED INTERRUPT
1347                                                            SEL2 = 10
1348                                                            SEL4 = CLOCK PERIOD
1349
1350                                                    - DCT11 INTERRUPTED WHILE CLOCK DISABLE
1351                                                            SEL2 = 12
1352
1353                                                    THEN THE HOST REPORTS CORRESPONDING ERROR.
1354
1355                                    MAINT MODE:     2
1356
1357                                    KMV ROUTINE:    11
1358
1359                                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1360                                                    ERROR 11        NO KMV11 RESPONSE
1361                                                    ERROR 25        INTERRRUPT OCCURS TOO EARLY
1362                                                    ERROR 26        COUNTER DECREMENTED BUT NO
1363                                                                    INTERRUPT
1364                                                    ERROR 27        COUNTER STOPPED
1365                                                    ERROR 30        KMV11 INTERRUPTED AT ILLEGAL VECTOR
1366                                                    ERROR 31        INTERRUPT WHILE CLOCK DISABLED
1367
1368
1369
1370
1371                                    7.17 TEST17
1372
```

```
1373                    ** - TEST OF TRANSMIT DATA IN INTERRUPT MODE
1374
1375                    DESCRIPTION:    THE HOST ASKS  DCT11 TO  TEST  DATA  TRANS-
1376                                    MISSION IN  INTERRUPT MODE.
1377                                    THE DCT11 TESTS ALL XMT RELATIVE
1378                                    CONTROL  BITS  AND INTERRUPTS WHILE THE RE-
1379                                    CEIVER IS DISABLED. IT TESTS LINE CLOCK COUN-
1380                                    TER TOO. THE LINE  CLOCK RATE IS PREVIOUSLY
1381                                    LOADED IN SEL2 BY THE HOST:
1382
1383                                            SEL2 = 226      FOR     72K
1384                                                 = 410      FOR     64K
1385                                                 = 444      FOR     56K
1386                                                 = 504      FOR     48K
1387                                                 = 1540     FOR     19.2K
1388                                                 = 3440     FOR     9.6K
1389                                                 = 24200    FOR     2.4K
1390
1391                                    THE HOST SELECTS SPEED BY MEANS OF P.TABLE.
1392                                    IN THE CASE OF ALL SPEED REQUESTED,THE TEST
1393                                    IS REPEATED FOR EACH SPEED.PERIODICALLY THE
1394                                    HOST POLLS SEL0 FOR ANY DCT11 RESPONSE:
1395
1396                                            SEL0 = 0        IF TEST OKE
1397                                            SEL0 = 100      IF ANY ERROR
1398
1399                                    IN THE LATTER CASE,THE FOLLOWING CSR'S CON-
1400                                    TAIN ERROR REPORTS, SUCH AS:
1401
1402                                            SEL2 = 2  ERROR 32
1403                                                 = 4  ERROR 33
1404                                                 = 6  ERROR 34
1405                                                 = 10 ERROR 40
1406                                                 = 12 ERROR 41
1407                                                 = 14 ERROR 42
1408                                                 = 16 ERROR 43
1409                                            SEL4 = INTERRUPT COUNT
1410                                    THEN, THE HOST REPORTS ERROR AND EXIT.
1411
1412                    MAINT MODE:     2
1413
1414                    KMV ROUTINE:    12
1415
1416                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1417                                    ERROR 11        NO KMV11 RESPONSE
1418                                    ERROR 32        LINE CLOCK COUNTER STOPPED
1419                                    ERROR 33        TIMEOUT WITHOUT INTERRUPT
1420                                    ERROR 34        INTERRUPT AT ILLEGAL VECTOR
1421                                    ERROR 40        UNEXPECTED INTERRUPT WHILE
1422                                                    XMT INTERRUPT DISABLE
1423                                    ERROR 41        NO INTERRUPT WHILE WAITING UNDERRUN
1424                                    ERROR 42        UNDERRUN INTERRUPT AT ILLEGAL VECTOR
1425                                    ERROR 43        UNDERRUN INTERRUPT WHILE NOT EXPECTED
1426
1427
1428                    7.18 TEST18
1429                    7.19 TEST19
```

```
1430                                    7.20 TEST20
1431                                    7.21 TEST21
1432                                    7.22 TEST22
1433
1434                                     ** - TEST OF DATA RECEPTION IN INTERRUPT MODE
1435
1436                                     DESCRIPTION:    THE HOST ASKS THE  DCT11 TO  TEST  DATA  RE-
1437                                                     CEPTION  BY  ITSELF. IN ORDER TO DO THAT, IT
1438                                                     PASSES TO THE DCT11 PARAMETERS SUCH AS:
1439                                                             SEL2 = LINE CLOCK RATE(AS DEFINED IN
1440                                                                    TEST16)
1441
1442                                                     THE TEST IS COMPOSED OF 5 SUBTESTS:
1443
1444                                                     - TEST 18
1445                                                     TEST OF INTERRUPTS ON NON-ZERO STATUS AND
1446                                                     DATA RECEPTION FOR ONE FRAME SENT,SUCH AS:
1447
1448                                                     ,--------,
1449                                                     )TSOM    )
1450                                                     )1       )
1451                                                     )2       )   >---
1452                                                     )3       )       )
1453                                                     )4       )       )
1454                                                     )5       )       )
1455                                                     )6       )       )
1456                                                     )1       )       )
1457                                                     )2       )       )
1458                                                     )TEOM    )       )
1459                                                     ---------        )
1460                                                     XMT FIFO         )
1461                                                                      )
1462                                                     ,--------,       )
1463                                                     )RSOM!1  )       )
1464                                                     )2       )   <---
1465                                                     )3       )
1466                                                     )4       )
1467                                                     )5       )
1468                                                     )6       )
1469                                                     )1       )
1470                                                     )REOM!2  )
1471                                                     ---------
1472                                                     RCV FIFO
1473
1474
1475
1476                                                     - TEST 19
1477                                                     TEST OF TWO FRAMES CONSECUTIVELY SENT BUT
1478                                                     ONLY THE SECOND ONE IS TAKEN INTO ACCOUNT.
1479
1480
1481                                                ,--------,
1482                                                )TSOM    )
1483                                                )1       )
1484                                                )2       )
1485                                                )3       )
1486                                                )4       )
                                                   )5       )
```

```
                    )TEOM    )   >--------------->          ---------
                    )TSOM    )                             )RSOM!1 )
                    )1       )                             )REOM!5 )
                    )2       )                             )RSOM!1 )
                    )3       )                             )2      )
                    )4       )                             )3      )
                    )5       )                             )4      )
                    )6       )                             )5      )
                    )1       )                             )6      )
                    )2       )                             )1      )
                    )TEOM    )                             )REOM!2 )
                    ---------                              ---------
                    XMT FIFO                               RCV FIFO


                            - TEST 20
                            AN ABORT CHARACTER IS RECEIVED DUE TO A TRANSMIT
                            UNDERRUN AND ANOTHER ONE DUE TO A REAL ONE TRANSMITTED.


                    ---------
                    )TSOM    )                             ---------
                    )1       )       ------>               )RSOM!1 )
                    )2       )                             )2      )
                    )3       )                             )3      )
                    )4       )                             )4      )
                    )5       )
                    )6       )
                    )1       )
                    THEN STOP
                    UNTIL UNDERRUN  ------>                RABO*REOM!4


                    )TSOM    )
                    )1       )       ------>
                    )2       )                             RSOM!1
                    )3       )                             RABO*REOM!1
                    )4       )
                    )TXAB    )
                    ---------
                    XMT FIFO


                            - TEST 21
                            TEST OF NON ZERO STATUS INTERRUPT IN CASE OF
                            CRC ERROR. THE COM DEVICE
                            IS INITIALIZED TO SEND A FLAG INSTEAD OF AN
                            ABORT(IDLE=1) ON TXAB SET


                    ---------
                    )TSOM    )                             ------------------
                    )1       )                             )RSOM!1           )
                    )2       )                             )2                )
                    )3       )                             )3                )
                    )4       ) --> FLAG -->                )ERRCHK!REOM 4    )
                    )5       )                             ------------------
                    )6       )
                    )TABO    )
                    ---------                              RCV FIFO
```

```
1544                                    XMT FIFO
1545
1546
1547
1548                                          - TEST 22
1549                                          TEST OF RECEIVE OVERRUN
1550
1551                                    ---------                    ---------
1552                                    )TSOM   )  >---------------> RSOM )1      )
1553                                    )1      )                    ROR  )2      )
1554                                    )2      )                         )3      )
1555                                    )3      )                         )4      )
1556
1557                                    UP TO 68 INCREMENTAL DATA            UP TO 64 DATA IN
1558
1559                                    )104    )                            )       )
1560                                    )TEOM   )                            ---------
1561                                    ---------
1562
1563
1564                                          PERIODICALLY THE HOST POLLS SEL0 FOR ANY DCT11
1565                                          RESPONSE:
1566                                                SEL0 = 0        FOR TEST OKE
1567                                                SEL0 = 100      IF ANY ERROR
1568                                          IN THE LATTER CASE SEL2 POINTS TO THE ERROR,
1569                                          POSSIBLE ERRORS ARE :
1570                                                SEL2 = 2        FOR ERROR 32
1571                                                     = 4        FOR ERROR 45
1572                                                     = 6        FOR ERROR 40
1573                                                     = 10       FOR ERROR 46
1574                                                     = 12       FOR ERROR 47
1575                                                     = 14       FOR ERROR 50
1576                                                     = 16       FOR ERROR 51
1577                                                     = 20       FOR ERROR 35
1578                                                     = 22       FOR ERROR 43
1579                                                SEL4 = OBTAINED STATUS&DATA
1580                                                SEL6 = EXPECTED ONE
1581                                                SEL10= EVENT FLAGS
1582
1583                                    MAINT MODE:     2
1584
1585                                    KMV ROUTINE:    13              FOR TEST 18
1586                                                    14              FOR TEST 19
1587                                                    15              FOR TEST 20
1588                                                    16              FOR TEST 21
1589                                                    17              FOR TEST 22
1590
1591                                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1592                                                    ERROR 11        NO KMV11 RESPONSE
1593                                                    ERROR 32        LINE CLOCK FAILURE
1594                                                    ERROR 40        TRANSMIT INTERRUPT WHILE DISABLED
1595                                                    ERROR 35        UNDERRUN INTERRUPT
1596                                                    ERROR 43        UNEXPECTED UNDERRUN INTERRUPT AFTER ACK
1597                                                    ERROR 45        XMT/RCV PROCESSING FAILS
1598                                                    ERROR 46        UNEXPECTED STATUS RECEIVED
1599                                                    ERROR 47        UNEXPECTED DATA RECEIVED
1600                                                    ERROR 50        RECEIVE INTERRUPT WHILE DISABLED
```

```
                                        ERROR 51        INTERRUPT AT ILLEGAL VECTOR


                7.23 TEST23
                7.24 TEST24
                7.25 TEST25

        TEST23: TEST OF DMA IN (TRX) TRANSFER USING A SHORT FRAME OF DATA
                THE FRAME USED IS THE PATTERN TABLE:
                                ALL 0
                                ALL 1
                                052525
                                125252
                                ROTATING 0
                                ROTATING 1

        TEST24: TEST OF DMA IN (TRX) TRANSFER USING A LONG FRAME OF DATA
                THE FRAME USED IS 1020. BYTES LONG AND CONTAINS THE FOLLOWING DATA:
                0,1,2...376,0,1,2...376...0,1,2...376

        TEST25:TEST OF NON EXISTENT MEMORY TEST DURING A TRANSMIT DMA TRANSFER
                THE NON EXISTENT MEMORY ADDRESS USED IS THE REVERVED ONE 760000.

         DESCRIPTION:    THE HOST LOADS THE FRAME PARAMETERS IN THE CSR.
                         THEN IT ACTIVATES THE FW ROUTINE TO TRANSFER
                         BY DMA THE FRAME INTO THE XMT  FIFO , TO
                         TRANSMIT IT,AND RECEIVE IT BACK. THE DCT11
                         CHECKS THE RECEIVED FRAME. IN ORDER TO  INIT
                         DMA TRANSFER THE  HOST PASSES PARAMETERS  TO
        THE DCT11, SUCH AS:

                SEL2 = LINE CLOCK RATE
                SEL4 = DMA IN BYTE COUNT(2'S COMP)
                SEL6 = DMA IN BUS ADDRESS(EVEN)
                SEL10= EXT ADDRESS
                SEL12= SUBTEST CODE

        NOTICE  THAT  DMA IN  IS WORD WIDE BUT, EACH
        WORD IS LOADED IN FIFO BYTE BY BYTE.
        PERIODICALLY THE HOST POLLS SEL0 FOR ANY RES
        PONSE FROM THE KMV11:

                SEL0 = 0 IF TEST CORRECTLY ENDED
                SEL0 = 100 IF ANY ERROR
        IN THE LATTER CASE:

                SEL2 =  2       FOR ERROR 32
                        4       FOR ERROR 45
                        6       FOR ERROR 54
                        10      FOR ERROR 55
                        12      FOR ERROR 56
                        14      FOR ERROR 46
                        16      FOR ERROR 57
                        20      FOR ERROR 50
                        22      FOR ERROR 51
                        24      FOR ERROR 35
```

```
                                        SEL10 =  FLAGWORD DESCRIBING PROCESSING
                                        SEL12 =  CURRENT TRX DMA BYTE COUNT
                                        SEL14 =  CURRENT TRX DMA BUFFER ADDRESS (LOW)
                                        SEL16 =  CURRENT TRX DMA BUFFER ADDRESS (HIGH)
                                    IF ERROR 46,50,57 :
                                        SEL4=  RCV STATUS AND DATA REGISTER
                                        SEL6=  EXPECTED VALUE

                                THEN THE HOST BUILDS  ERROR REPORT  AND EXIT

                            MAINT MODE:    2

                            KMV ROUTINE:   20

                        REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 11        NO KMV11 RESPONSE
                                        ERROR 32        LINE COUNTER ERROR
                                        ERROR 35        UNDERRUN INTERRUPT OCCURS
                                        ERROR 45        TRX/RCV ERROR
                                        ERROR 46        UNEXPECTED STATUS RECEIVED
                                        ERROR 50        UNEXPECTED RCV INT. WHILE DISABLED
                                        ERROR 51        INTERRUPT AT ILLEGAL VECTOR
                                        ERROR 54        UNEXPECTED END OF RCV DMA INT.
                                        ERROR 55        END OF TRX DMA WITHOUT BIT "BCNTIN" SET
                                        ERROR 56        TIMEOUT DURING TRX DMA
                                        ERROR 57        DATA COMPARE ERROR DURING TRX DMA


                            7.26 TEST26
                            7.27 TEST27
                            7.28 TEST28

                            TEST26: TEST OF DMA OUT (RCV) TRANSFER USING A SHORT FRAME OF DATA
                            TEST27: TEST OF DMA OUT (RCV) TRANSFER USING A LONG FRAME OF DATA
                            TEST28: TEST NON EXISTENT MEMORY ADDRESS DURING A OUT DMA TRANSFER

                             DESCRIPTION:   THE HOST PREPARES A  SPACE IN ITS MEMORY FOR
                                            THE TRANSFERRED FRAME. THEN, THE HOST ASKS
                                            THE DCT11 FOR PROCESSING. IN ORDER TO INIT
                                            DMA OUT, THE HOST PASSES PARAMETERS TO THE
                                            KMV, SUCH AS:
                                                    SEL2 = LINE CLOCK RATE
                                                    SEL4 = DMA OUT BYTE COUNT(2'S COMP.)
                                                    SEL6 = RECEIVE BUFFER ADDRESS(EVEN)
                                                    SEL10= EXT. ADDRESS
                                                    SEL12= SUBTEST CODE (0 OR 2)

                                            THE DCT11 SENDS , RECEIVES AND TRANSFERS BY
                                            DMA THE SAME FRAMES AS NOTICE IN TESTS 23,24,25.
                                            DMA OUT IS BYTE WIDE.
                                            PERIODICALLY THE HOST POLLS SEL0 FOR ANY RES
                                            PONSE FROM THE KMV11:

                                                    SEL0 = 0 IF TEST OKE
                                                    SEL0 = 100 IF ANY ERROR
```

```
1715                                    IN THE FORMER CASE, THE HOST CHECK THE FRAME
1716                                    LOADED BY DMA AND REPORT ERROR IF A DATA
1717                                    COMPARE ERROR.
1718                                    IN THE LATTER CASE,
1719                                          SEL2 =  2       FOR ERROR 32
1720                                                  4       FOR ERROR 45
1721                                                  6       FOR ERROR 61
1722                                                  10      FOR ERROR 62
1723                                                  12      FOR ERROR 63
1724                                                  14      FOR ERROR 46
1725                                                  16      FOR ERROR 51
1726                                                  20      FOR ERROR 35
1727                                          SEL4 =  RCV STATUS AND DATA REGISTER (ERROR 46)
1728                                          SEL6 =  EXPECTED STATUS AND DATA REGISTER  (ERROR 46)
1729                                          SEL10=  FLAGWORD
1730                                          SEL12=  CURRENT RCV BYTE COUNT
1731                                          SEL14 = CURRENT RCV BUFFER ADDRESS (LOW)
1732                                          SEL16 = CURRENT RCV BUFFER ADDRESS (HIGH)
1733                                    THEN THE HOST BUILDS ERROR REPORTS DEPENDING
1734                                    ON SEL0 THEN EXIT.
1735
1736                            MAINT MODE:     2
1737
1738                            KMV ROUTINE:    21
1739
1740                            REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1741                                            ERROR 11        NO KMV11 RESPONSE
1742                                            ERROR 35        UNDERRUN INTERRUPT OCCURS
1743                                            ERROR 45        TRX/RCV ERROR
1744                                            ERROR 46        UNEXPECTED RECEIVED STATUS
1745                                            ERROR 61        UNEXPECTED END OF TRX DMA
1746                                            ERROR 62        RCV BYTE COUNT OVERFLOW
1747                                            ERROR 63        TIMEOUT DURING RCV DMA
1748                                            ERROR 51        INTERRUPT AT ILLEGAL VECTOR
1749
1750                                    7.29 TEST29
1751                                    7.30 TEST30
1752                                    7.31 TEST31
1753                                    7.32 TEST32
1754
1755                            TESTS OF DMA TRANSFER IN BOTH DIRECTIONS
1756
1757                             DESCRIPTION:   THE HOST ASKS THE  DCT11 FOR  INITIALIZING A
1758                                            DMA TRANSFER IN BOTH DIRECTIONS .
1759                                            IN ORDER TO RUN TEST, THE HOST PASSES TO THE
1760                                            DCT11 PARAMETERS, SUCH AS:
1761
1762                                                    SEL2 = LINE CLOCK RATE
1763                                                    SEL4 = DMA TRX BYTE COUNT(2'S COMP.)
1764                                                    SEL6 = DMA TRX BUS ADDRESS(EVEN)
1765                                                    SEL10= DMA TRX IN EXT. ADDRESS
1766                                                    SEL12= DMA RCV BYTE COUNT
1767                                                    SEL14= DMA RCV OUT BUS ADDRESS
1768                                                    SEL16= DMA RCV OUT EXT. ADDRESS
1769                                                         BIT15 OF SEL16 SET IF SEL20 NOT VALID (MEANS THE
1770                                                                DATA BYTE MUST BE IGNORED WHEN A STATUS IS RECEIVED.
1771                                                    SEL20= LAST EXPECTED DATA TO BE RECEIVED
```

```
1772
1773                                    THEN THE HOST PERIODICALLY POLLS  SEL0 FOR A
1774                                    RESPONSE FROM THE DCT11:
1775                                            SEL0 = 0 IF TEST COMPLETED
1776                                            SEL0 = 100 IF ANY ERROR
1777                                    IN THE FORMER CASE THE HOST COMPARES THE XMT
1778                                    TABLE WITH THE RCV ONE AND REPORT  ERROR  64
1779                                    IF ANY DATA COMPARE ERROR.
1780                                    IN THE LATTER CASE:
1781                                            SEL2 = 2        FOR ERROR 32
1782                                                 = 4        FOR ERROR 45
1783                                                 = 6        FOR ERROR 56
1784                                                 = 10       FOR ERROR 62
1785                                                 = 12       FOR ERROR 63
1786                                                 = 14       FOR ERROR 55
1787                                                 = 16       FOR ERROR 46
1788                                                 = 20       FOR ERROR 51
1789                                                 = 22       FOR ERROR 35
1790
1791                                                SEL4 = RECEIVED STATUS&DATA REGISTER (ERROR 46)
1792                                                SEL6 = EXPECTED STATUS&DATA REGISTER (ERROR 46)
1793                                                SEL10= FLAGWORD
1794                                                SEL12= TRX DMA BYTE COUNT
1795                                                SEL14= TRX DMA BUS ADDRESS
1796                                                SEL16= TRX DMA EXT. ADDRESS
1797                                                SEL20= RCV DMA BYTE COUNT
1798                                                SEL22= RCV DMA BUS ADDRESS
1799                                                SEL24= RCV DMA EXT. ADDRESS
1800                                    THEN THE HOST BUILDS REPORT AND EXIT
1801
1802                                    MAINT MODE:     2
1803
1804                                    KMV ROUTINE:    22
1805
1806                                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1807                                                    ERROR 11        NO KMV11 RESPONSE
1808                                                    ERROR 32        LINE COUNTER ERROR WHE  STARTED
1809                                                    ERROR 35        UNDERRUN INTERRUPT OCCURS
1810                                                    ERROR 45        TRX/RCV ERRORS
1811                                                    ERROR 46        UNEXPECTED STATUS RECEIVED
1812                                                    ERROR 51        INTERRUPT AT ILLEGAL VECTOR
1813                                                    ERROR 55        BIT "BCNTIN" NOT SET AT END OF TRX DMA
1814                                                    ERROR 56        Q-BUS TIMEOUT DURING TRX DMA
1815                                                    ERROR 62        BYTE COUNT OVERFLOW DURING RCV DMA
1816                                                    ERROR 63        Q-BUS TIMEOUT DURING RCV DMA
1817                                                    ERROR 64        DATA COMPARE ERROR
1818
1819
1820                                    TEST29:
1821                                     THE FIRST TEST OF THIS CLASS CORRESPONDS TO A 11 BYTES LONG TRANSMIT BUFFER
1822                                     AND A 15 BYTES LONG RECEIVE BUFFER.
1823
1824                                    TEST30:
1825                                     THE SECOND TEST OF THIS CLASS CORRESPONDS TO A 10 BYTES LONG TRANSMIT BUFFER
1826                                     AND A 8 BYTES LONG RECEIVE BUFFER.
1827
1828                                    TEST31:
```

```
1829                    THE THIRD TEST OF THIS CLASS CORRESPONDS TO A 10 BYTES LONG TRANSMIT BUFFER
1830                    AND A 6 BYTES LONG RECEIVE BUFFER. THIS TEST PERMITS TO TEST THE BUFFER
1831                    OVERLFOW.
1832
1833                   TEST32:
1834                    THE FOURTH TEST OF THIS CLASS CORRESPONDS TO A LONG DMA TRANSFER.
1835                    1020. BYTES WILL BE TRANSMITTED AND RECEIVED.
1836
1837                   7.33 TEST33
1838
1839                    TEST OF DMA TRANSFER IN BOTH DIRECTIONS AND IN HIGHER
1840                    PART OF THE HOST MEMORY TO CHECK THE EXTENDED ADDRESS COUNTER
1841
1842                    INPUT PARAMETERS FOR FW ROUTINE:
1843                           SEL2 = LINE CLOCK RATE
1844                           SEL4 = TRX DMA BYTE COUNT (IN 2'S COMP FORM)
1845                           SEL6 = TRX DMA BUFFER ADDRESS (LOW)
1846                           SEL10= TRX DMA BUFFER ADDRESS (HIGH)
1847                           SEL12= RCV DMA BYTE COUNT (IN 2'S COMP FORM)
1848                           SEL14= RCV DMA BUFFER ADDRESS (LOW)
1849                           SEL16= RCV DMA BUFFER ADDRESS (HIGH)
1850                                  BIT 15 SET IF SEL20 MUST NOT BE TAKEN INTO ACCOUNT IN FW TEST22
1851                           SEL20= LAST EXPECTED DATA TO BE RECEIVED
1852
1853                    OUTPUT:
1854                           SEL0 = 0 IF SUCCESS
1855                           SEL0 = 100 IF ANY ERROR
1856                                  WITH SEL2 =
1857
1858                                         SEL12= TRX DMA BYTE COUNT
1859                                         SEL14= TRX DMA BUFFER ADDRESS (LOW)
1860                                         SEL16= TRX EXT ADDRESS
1861                                         SEL20= RCV DMA BYTE COUNT
1862                                         SEL22= RCV DMA BUFFER ADDRESS (HIGH)
1863                                         SEL24= RCV DMA EXT. BUFFER ADDRESS
1864
1865                           MAINT MODE:    2
1866
1867                           KMV ROUTINE:   22
1868
1869                          7.34 TEST34
1870
1871                   TEST OF MODEM SIGNALS (EXTERNAL LOOP-BACK MODE)
1872
1873                   DESCRIPTION:   THIS TEST CAN BE RUN IN THE CASE OF EXTERNAL
1874                                  LOOP ONLY. AN ERROR MESSAGE IS PRINTED IF NO
1875                                  LOOP-BACK CONNECTOR PLUGGED.
1876                                  OTHERWISE, THE HOST ASKS THE DCT11 TO SET-UP
1877                                  MODEM CONTROLS BY MEANS OF THE 'MAILBOX' ROU
1878                                  TINE WITH THE FOLLOWING INPUTS:
1879                                         SEL2 = PORT B ADDRESS
1880                                         SEL4 = MODEM CONTROL BIT(S)
1881                                         SEL6 = 6 FOR WRITE BYTE
1882
1883                                  THEN TO READ LOOP BACK RESULT BY MEANS OF THE
1884                                  'MAILBOX ROUTINE WITH THE FOLLOWING INPUTS:
1885                                         SEL2 = PORT A ADDRESS
```

```
                                        SEL4 = 377
                                        SEL6 = 4 FOR READ BYTE
                                THE HOST POLLS SEL0 FOR "MAILBOX" ROUTINE JOB
                                COMPLETED:
                                        SEL0 = 0 FOR JOB COMPLETE
                                IN THE LOOP BACK RESULT CASE,
                                        SEL4 = EXPCTED LOOP BACK RESULT
                                THEN THE HOST TESTS/REPORTS AND EXITS.

                                NOTICE, THAT MODEM SIGNALS  ARE LOOPED, SUCH
                                AS:

                                CCITT 108/2 --------.
                                                    )
                                CCITT 105   ----.   )
                                               )    )
                                CCITT 107   <---)---'
                                               )
                                CCITT 106   <---)
                                               )
                                CCITT 109   <---'

                                MODEM CCITT 108/2 AND 105 WILL BE SET ONE BY
                                ONE THEN TOGETHER.

MAINT MODE:     2

KMV ROUTINE:    23

REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
                ERROR 11        NO KMV11 RESPONSE
                ERROR 74        MODEM SIGNAL BIT SETTING NOT
                                CORRECT
                ERROR 75        NO LOOP BACK CONNECTOR

        7.35 TEST35

THIS TEST LOADS A ROUTINE IN THE RAM AND CHECKS IF ITS
EXECUTION IS CORRECT.
THE ROUTINE LOADED IS THE FW TEST06

DESCRIPTION:    THE HOST ASKS THE DCT11 FOR TESTING ALL R/W
                REGISTERS BY ITSELF. PERIODICALLY, THE HOST
                POLLS SEL0 FOR ANY DCT11 RESPONSE:
                        SEL0 = 0        FOR TEST OKE
                        SEL0 = 100      FOR ANY ERROR
                IN THE LATTER CASE, THE FOLLOWING REGISTERS
                CONTAINS ERROR REPORTS, SUCH AS:
                        SEL2 = 2        BYTE COUNT OUT REG.
                             = 4        BUS ADDRESS OUT REG.
                             = 6        EXT ADDRESS OUT REG.
                             = 10       BYTE COUNT IN REG.
                             = 12       BUS ADDRESS IN REG.
                             = 14       EXT ADDRESS IN REG.
                        SEL4 = CURRENT PATTERN
                        SEL6 = BAD WORD
```

```
1943                    MAINT MODE:     2
1944
1945                    KMV ROUTINE:    27
1946
1947                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1948                                    ERROR 11        NO RESPONSE FROM THE KMV11
1949                                    ERROR 17        DATA COMPARE ERROR ON KMV11
1950                                                    INTERNAL R/W REGISTER
1951
1952                            7.36 TEST36
1953
1954                    RECEIVE DATA TEST USING THE POLLING FACILITY INSTEAD OF INTERRUPT
1955
1956                    DESCRIPTION:
1957                                    THIS TEST SELECT IN THE NATIVE FIRMWARE THE TEST30,
1958                                    WHICH TESTS TRANSMISSION AND PECEPTION OF DATA USING
1959                                     THE POLLING FACILITY INSTEAD OF INTERRUPT.
1960                                    PERIODICALLY THE HOST POLLS SEL0 FOR ANY DCT11
1961                                    RESPONSE:
1962                                            SEL0 = 0        FOR TEST OKE
1963                                    SEL0 = 100      IF ANY ERROR
1964                                    IN THE LATTER CASE SEL2 POINTS TO THE ERROR,
1965                                    POSSIBLE ERRORS ARE :
1966                                            SEL2 = 2        FOR ERROR 32
1967                                                 = 4        FOR ERROR 45
1968                                                 = 6        FOR ERROR 40
1969                                                 = 10       FOR ERROR 46
1970                                                 = 12       FOR ERROR 47
1971                                                 = 14       FOR ERROR 50
1972                                                 = 16       FOR ERROR 51
1973                                                 = 20       FOR ERROR 35
1974                                                 = 22       FOR ERROR 43
1975                                            SEL4 = OBTAINED STATUS&DATA
1976                                            SEL6 = EXPECTED ONE
1977                                            SEL10= EVENT FLAGS
1978
1979                    MAINT MODE:     2
1980
1981                    KMV ROUTINE:    30
1982
1983
1984                    REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
1985                                    ERROR 11        NO KMV11 RESPONSE
1986                                    ERROR 32        LINE CLOCK FAILURE
1987                                    ERROR 40        TRANSMIT INTERRUPT WHILE DISABLED
1988                                    ERROR 35        UNDERRUN INTERRUPT
1989                                    ERROR 43        UNEXPECTED UNDERRUN INTERRUPT AFTER ACK
1990                                    ERROR 45        XMT/RCV PROCESSING FAILS
1991                                    ERROR 46        UNEXPECTED STATUS RECEIVED
1992                                    ERROR 47        UNEXPECTED DATA RECEIVED
1993                                    ERROR 50        RECEIVE INTERRUPT WHILE DISABLED
1994                                    ERROR 51        INTERRUPT AT ILLEGAL VECTOR
1995
1996                            7.37 TEST37
1997
1998                    LONG RAM TEST
1999
```

```
                        THIS TEST DURATION IS ABOUT 1H15

                        DESCRIPTION:    THE HOST ASKS THE DCT11 FOR TESTING THE RAM
                                        BY ITSELF. PERIODICALLY THE HOST CHECKS THE
                                        REGISTER SEL0 FOR ANY DCT11 RESPONSE:
                                                SEL0 = 0        IF TEST SUCCEEDED
                                                SEL0 = 100      IF ANY ERROR
                                        IN THE LATTER CASE, THE FOLLOWING REGISTERS
                                        CONTAINS THE ERROR REPORTS AS FOLLOW:

                                        - IF THE DCT11 CAN'T CLEAR ALL THE RAM
                                                SEL2 = 2
                                                SEL4 = FIRST BAD WORD ADDRESS
                                                SEL6 = FIRST BAD WORD

                                        - IF DATA COMPARE ERROR AT LOCATION TESTED
                                                SEL2 = 4
                                                SEL4 = RAM ADDRESS
                                                SEL6 = BAD WORD
                                                SEL10= PATTERN

                                        - IF WRITTEN PATTERN IN ONE LOCATION PRODU-
                                          CES FALSE DATA IN OTHER LOCATIONS
                                                SEL2 = 6
                                                SEL4 = FIRST BAD WORD ADDRESS
                                                SEL6 = FIRST BAD WORD
                                                SEL10= PATTERN
                                                SEL12= LOCATION UNDER TEST

                        MAINT MODE:     2

                        KMV ROUTINE:    05

                        REPORTS:        ERROR 2         MASTER CLEAR FAILS TO RESET
                                        ERROR 11        NO ANSWER FROM THE KMV11
                                        ERROR 14        ALL THE RAM CAN'T BE CLEARED
                                        ERROR 15        DATA COMPARE ERROR ON RAM AT
                                                        LOCATION UNDER TEST
                                        ERROR 16        DATA COMPARE ERROR ON RAM AT
                                                        LOCATION NOT UNDER TEST
```

```
2041                    8.0 ERROR INFORMATION
2042
2043
2044                    8.1 ERROR REPORTING
2045
2046                    ERRORS ARE REPORTED BY THE PROGRAM AS  THEY  OCCUR  (IF  NOT
2047                    INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR
2048                    ERROR REPORT FORMAT, AND CONSISTS OF A  DESCRIPTION  OF  THE
2049                    ERROR,  THE  TEST  NUMBER,  SUBTEST  NUMBER, PC OF THE ERROR
2050                    CALL,  DEVICE  ADDRESS,  AND  BASIC ERROR INFORMATION (EXTEN
2051                    DED ERROR INFORMATION CAN BE SUPPRESSED BY SETTING THE FLAG
2052                    SWITCH IXE).
2053
2054                    THE FOLLOWING EXAMPLES PROVIDE TYPICAL ERROR REPORTS:
2055
2056                    VKMHA0 HRD ERR 00011 ON UNIT 00 TST 007 SUB 000 PC: 032164
2057
2058                    NO ANSWER FROM THE KMV11-C
2059
2060
2061                    VKMHA0 HRD ERR 00016 ON UNIT 00 TST 013 SUB 000 PC: 031010
2062
2063                    DATA COMPARE ERROR ON RAM AT LOCATION NOT UNDER TEST
2064                    LOCATION UNDER TEST = XXXXXX    FOR PATTERN = XXXXXX
2065
2066                    BAD VALUE = XXXXXX      AT LOCATION = XXXXXX
2067
2068                    NOTICE THAT THE REPORT MAY BE MORE EXTENSIVE AND REQUIRE
2069                    ADDITIONAL DATA TO BE REPORTED.
```

```
2071
2072
2073
2074
2075
2076                    9.0     HISTORY
2077
2078                    - VERSION 01, REVISION A        DESIGN STARTED ON MAY 82
2079
2080                    - VERSION 01, REVISION B        27-MAY-83
2081
2082                    -VERSION  01, REVISION C        01-JULY-83
2083                    @
```

```
                                .TITLE VKMHA0 KMV11C STATIC DIAG
                                .IDENT /V01.05/

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


                        000000          HELP=0
000000                  .ENABL  ABS,AMA
                        002000          .=2000


                                .MCALL  SVC
002000                          SVC                             ; INITIALIZE SUPERVISOR MACROS


002000                          BGNMOD  VKMHA0

                        177777  $LSTIN= -1
                        177777  $LSTTAG= -1
                        177777  SVCINS= -1      ; LIST INSTRUCTIONS, SHIFTED RIGHT
                        177777  SVCTST= -1      ; LIST TEST TAGS, SHIFTED RIGHT
                        177777  SVCSUB= -1      ; LIST SUBTEST TAGS, SHIFTED RIGHT
                        177777  SVCGBL= -1      ; LIST GLOBAL TAGS, SHIFTED RIGHT
                        177777  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.
```

```
                        .SBTTL  PROGRAM HEADER
                        ;++
                        ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                        ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                        ;--

002000                          POINTER BGNSW,BGNAU,BGNDU,BGNSETUP

        000000          HELP=0

                        .IF NE HELP
                        ;*****************************************************************************
                        ;       IF ANY OPTIONAL POINTERS ARE TO BE USED IN THE "HEADER", CHANGE
                        ;       "POINTER" TO CONTAIN THE CORRECT ARGUMENTS.  IF ALL OPTIONAL
                        ;       POINTERS ARE TO BE USED, CHANGE "POINTER" TO BE "POINTER ALL".
                        ;       IF NO ARGUMENTS ARE USED ,CHANGE "POINTER" TO BE "POINTER NONE".
                        ;       ARGUMENT            OPTION
                        ;       --------            ------
                        ;       BGNRPT              REPORT CODE
                        ;       BGNSW               SOFTWARE TABLE
                        ;       BGNSFT              SOFTWARE TABLE QUESTIONS
                        ;       BGNAU               ADD CODE
                        ;       BGNDU               DROP CODE
                        ;       ERRTBL              ERROR TABLE
                        ;       BGNSETUP            ASSEMBLED P-TABLE
                        ;*****************************************************************************
                        .ENDC

002000                          HEADER VKMHA0,A,0,240..,0,PRI07    ; /V01.04/ ADD PRIORITY

                        .IF NE HELP
                        ;*****************************************************************************
                        ;       CHANGE THE "HEADER" TO CONTAIN THE PROPER ARGUMENTS.
                        ;       ARGUMENTS ARE: NAME,REV,PATCH,LONGEST TEST TIME,TYPE
                        ;       WHERE "TYPE" = 0 FOR SEQUENTIAL DIAGNOSTIC AND = 1
                        ;       FOR EXERCISER. THERE ALSO AN OPTIONAL SIXTH ARGUMENT
                        ;       WHICH SPECIFIES THE PROCESSOR PRIORITY TO BE SET WHEN
                        ;       STARTING THE DIAGNOSTIC.(DEFAULT IS 0).
                        ;*****************************************************************************
                        .ENDC
```

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

002122                                  BGNPROT

002122  177777                         -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
002124  177777                         -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
002126  177777                         -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER


                                .IF NE HELP
                                ;*****************************************************************************************************
                                ;       INSERT BYTE OFFSET FOR DATA NOTED IN COMMENTS ABOVE.  (OFFSET
                                ;       REFERS TO THE NUMBER OF BYTES FROM THE BEGINNING OF A PTABLE
                                ;       ENTRY TO THE ITEM IN QUESTION.)  IF THE PARTICULAR
                                ;       ITEM DOES NOT APPLY, LEAVE ENTRY AS -1.  WHEN THE RUNTIME
                                ;       SERVICES EXECUTES A GPHARD, IT USES THESE OFFSETS (IF NOT
                                ;       SET TO -1) TO GET THE ITEMS AND COMPARE WITH THOSE SAVED
                                ;       IN THE XXDP+ MONITOR.  IF THE UNIT BEING REQUESTED MATCHES THE
                                ;       LOAD DEVICE, THE RUNTIME SERVICES RETURN AN INCOMPLETE FLAG ON
                                ;       THE GPHARD.
                                ;*****************************************************************************************************
                                .ENDC


002130                                  ENDPROT
```

```
                        .SBTTL  DISPATCH TABLE

                        ;//////////////////////////////////////////////////////////////////////////////
                        ;/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                        ;/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                        ;//////////////////////////////////////////////////////////////////////////////

002130                          DISPATCH 37.

                        .IF NE HELP
                        ;*****************************************************************************
                        ;       CHANGE THE ARGUMENT OF "DISPATCH" TO BE THE
                        ;       NUMBER OF HARDWARE TESTS IN YOUR PROGRAM.
                        ;*****************************************************************************
                        .ENDC
```

```
                                .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.
                                ;/ AND IS USED AS A " TEMPLATE" FOR BUILDING THE P-TABLE
                                ;///////////////////////////////////////////////////////////////////////////

                                .ENABL  AMA
002244                                  BGNHW   DFPTBL

                                .IF NE HELP
                                ;*****************************************************************************
                                ;       PLACE YOUR DEFAULT HARDWARE P-TABLE HERE.  THE VALUES AND
                                ;       SIZE WILL BE USED AS A "TEMPLATE" FOR CREATING ACTUAL P-TABLE
                                ;       ENTRIES AND THE DEFAULT VALUES IN THE OPERATOR DIALOGUE.
                                ;       THE ACTUAL P-TABLE BUILT AT RUNTIME IS STORED IN SUPERVISOR
                                ;       SPACE.
                                ;*****************************************************************************
                                .ENDC

002246  177000                          .WORD   177000          ;KMV11-C CSR ADDRESS
002250  000300                          .WORD   300             ;KMV11-C VECTOR ADDRESS IN
002252  004000                          .WORD   4000            ;INTERRUPT PRIORITY LEVEL
002254  000001                          .WORD   1               ;TEST LOOPBACK CONNECTOR INSTALLED FLAG
002256  000007                          .WORD   7               ;CONTAINS BAUD RATE INDICATOR
                                                                ;0=2.4K , 1=9.6K , 2=19.2K , 3=48K , 4=56K
                                                                ;5=64K , 6=72K , 7=ALL
002260  000000                          .WORD   0               ; LONG RAM TEST TO BE RUN

002262                                  ENDHW
```

```
170
171                             .IF NE HELP
172                             ;*************************************************************************
173                             ;       PLACE YOUR SOFTWARE P-TABLE HERE.  THIS TABLE
174                             ;       IS OPTIONAL.  THIS TABLE, UNLIKE THE HARDWARE TABLE, WILL CONTAIN
175                             ;       THE ACTUAL VALUES ENTERED BY THE OPERATOR.
176                             ;*************************************************************************
177                             .ENDC
178
179
180
181
182
```

```


                                   .SBTTL  GLOBAL EQUATES SECTION


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



                                   .IF NE HELP
                                   ;¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢
                                   ;     NOTE:  THE ASSEMBLER DIRECTIVE '.GLOBL' SHOULD NOT BE USED ANYWHERE IN
                                   ;            YOUR DIAGNOSTIC PROGRAM.  INSTEAD, USE THE DOUBLE-EQUATE (==) OR
                                   ;            DOUBLE-COLON (::) TO DEFINE GLOBAL VALUES.  HELP-MESSAGES GIVEN
                                   ;            BELOW EXPLAIN WHERE TO USE THE DOUBLE-EQUATE AND DOUBLE-COLON
                                   ;            DIRECTIVES.
                                   ;¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢
                                   .ENDC


                                   .IF NE HELP
                                   ;¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢
                                   ;       THE "EQUALS" MACRO DEFINES SOME COMMON SYMBOLS: BIT DEFINITIONS,
                                   ;       SUPERVISOR EVENT FLAGS, SUPERVISOR FLAGS AND PROCESSOR PRIORITIES.
                                   ;
                                   ;       ADD TO THIS SECTION YOUR EQUATED VALUES THAT ARE USED IN MORE
                                   ;       THAN ONE TEST.  EQUATES THAT ARE USED IN ONLY ONE TEST SHOULD BE
                                   ;       PLACED AT THE FRONT OF THE TEST.
                                   ;
                                   ;       EQUATES THAT ARE ADDED TO THIS SECTION MUST BE DEFINED WITH A
                                   ;       DOUBLE-EQUATE (==); THIS WILL MAKE THEM GLOBAL EQUATES.
                                   ;       FOR EXAMPLE, AN ERROR STATUS BIT COULD BE DEFINED AS ERR==BIT15.
                                   ;¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢¢
                                   .ENDC
                                   .LIST   ME
    002262                                 EQUALS
                                   ;
                                   ; BIT DIFINITIONS
                                   ;
                100000             BIT15== 100000
                040000             BIT14== 40000
                020000             BIT13== 20000
                010000             BIT12== 10000
                004000             BIT11== 4000
                002000             BIT10== 2000
                001000             BIT09== 1000
                000400             BIT08== 400
                000200             BIT07== 200
                000100             BIT06== 100
                000040             BIT05== 40
                000020             BIT04== 20
                000010             BIT03== 10
                000004             BIT02== 4
                000002             BIT01== 2
```

```
000001                  BIT00== 1
                        ;
001000                  BIT9==  BIT09
000400                  BIT8==  BIT08
000200                  BIT7==  BIT07
000100                  BIT6==  BIT06
000040                  BIT5==  BIT05
000020                  BIT4==  BIT04
000010                  BIT3==  BIT03
000004                  BIT2==  BIT02
000002                  BIT1==  BIT01
000001                  BIT0==  BIT00
                        ;
                        ; EVENT FLAG DEFINITIONS
                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                        ;
000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                        ;
                        ;
                        ; PRIORITY LEVEL DEFINITIONS
                        ;
000340                  PRI07== 340
000300                  PRI06== 300
000240                  PRI05== 240
000200                  PRI04== 200
000140                  PRI03== 140
000100                  PRI02== 100
000040                  PRI01== 40
000000                  PRI00== 0
                        ;
                        ;OPERATOR FLAG BITS
                        ;
000004                  EVL==     4
000010                  LOT==    10
000020                  ADR==    20
000040                  IDU==    40
000100                  ISR==   100
000200                  UAM==   200
000400                  BOE==   400
001000                  PNT==  1000
002000                  PRI==  2000
004000                  IXE==  4000
010000                  IBE== 10000
020000                  IER== 20000
040000                  LOE== 40000
100000                  HOE== 100000
223                     .NLIST  ME
224
225                     ;;*****************************************************************************************
226                     ;* PROGRAM EVENT FLAG DEFINITIONS
227                     ;;*****************************************************************************************
228
229                     ;
```

```
                        ; CSR OFFSETS DEFINITIONS
                        ;
000000                  SEL0    = 0
000002                  SEL2    = SEL0+2
000003                  SEL3    = SEL2+1
000004                  SEL4    = SEL2+2
000006                  SEL6    = SEL4+2
000010                  SEL10   = SEL6+2
000012                  SEL12   = SEL10+2
000014                  SEL14   = SEL12+2
000016                  SEL16   = SEL14+2
000020                  SEL20   = SEL16+2
000022                  SEL22   = SEL20+2
000024                  SEL24   = SEL22+2
000026                  SEL26   = SEL24+2
000030                  SEL30   = SEL26+2
000032                  SEL32   = SEL30+2
000034                  SEL34   = SEL32+2
000036                  SEL36   = SEL34+2

000020                  CSRLEN  = 16.                   ;LENGTH OF CSR IN WORDS
                        ;
                        ; SEL0 BIT DEFINITIONS
                        ;
100000                  RUN     = BIT15         ;RUN BIT
040000                  MCLR    = BIT14         ;MASTER CLEAR BIT
020000                  WRITE   = BIT13         ;WRITE BIT
010000                  TSTNUM  = BIT12         ;THIS BIT IS SET TO VALIDATE THE TEST NUMBER IN BSEL0
004000                  MAINT2  = BIT11         ;MODE MAINTENANCE 2
002000                  READ    = BIT10         ;READ BIT
001000                  MAINT1  = BIT09         ;MODE MAINTENANCE 1
000400                  ERR     = BIT08         ;ERROR BIT
```

```
                        .SBTTL  GLOBAL DATA SECTION

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

                        .IF NE HELP
                        ;*******************************************************************************
                        ;     INSERT INTO THE <> THE NAMES OF THE DEVICES TO BE TESTED.
                        ;*******************************************************************************
                        .ENDC


                        ;;*****************************************************************************
                        ;* STORAGE FOR DEVICE REGISTERS
                        ;;*****************************************************************************
002262                          DESCRIPT        <VKMHA0 KMV11-C LOGIC  DIAGNOSTIC>


                        .IF NE HELP
                        ;*******************************************************************************
                        ;       THE ERRTBL MACRO IS REQUIRED IF YOU INTEND TO REPORT ERRORS USING
                        ;       THE "ERROR" MACRO.  THE ERRTBL MACRO EXPANDS INTO FOUR WORDS THAT
                        ;       ARE USED BY THE RUNTIME SERVICES DURING AN ERROR CALL: ERROR TYPE,
                        ;       ERROR NUMBER, ADDRESS OF ERROR MESSAGE AND ADDRESS OF MESSAGE
                        ;       BLOCK.  THERE MUST BE ONLY ONE ERRTBL IN ANY PROGRAM.  THIS SECTION
                        ;       IS OPTIONAL.  REMOVE IF IT IF YOU ARE NOT GOING TO USE THE ERROR
                        ;       MACRO.  CHANGE THE POINTER MACRO TO REFLECT THIS SECTION'S DEL-
                        ;       ETION IF YOU REMOVE IT.
                        ;*******************************************************************************
                        .ENDC

002324                          ERRTBL
002324  000000          ERRTYP::        .WORD   0
002326  000000          ERRNBR::        .WORD   0
002330  000000          ERRMSG::        .WORD   0
002332  000000          ERRBLK::        .WORD   0


                        ;*****************************************************************************
                        ;* MISCELLANEOUS STORAGE
                        ;*****************************************************************************

002334  000000          LOGDEV: 0                       ;LOGICAL DEVICE NUMBER
002336  000000          PSTACK: 0                       ;BASE LEVEL PROGRAM STACK POINTER
002340  000000          FTIME:  0                       ;FIRST PASS FLAG
002342  000000          FLGP11: 0                       ;FIRST PASS FLAG WORD USED IN TEST11
002344  000000          SAVE4:  0                       ;SAVE INT. VECTOR 004
002346  000000          SAVE6:  0
002350  000000          GOOD:   0                       ;POINT TO GOOD DATA
002352  000000          BAD:    0                       ;POINT TO BAD DATA
002354  000000          SPADDR: 0                       ;STACK ADDRESS IN CASE OF ERROR 0
002356  000000          INTFLG: 0                       ;INTERRUPT FLAG
```

```
                                DATA:   0                       ;POINTER TO CURRENT PATTERN
                                RATE:   0                       ;POINT TO THE CURRENT LINE CLOCK RATE
                                NXMTST: 0                       ;FLAGWORD SET TO 1 WHEN NXM TEST IS RUNNING
                                LCLOOP: 0                       ;LOOPBACK CONNECTOR INDICATOR
                                REG1:   0
                                REG2:   0
                                REG3:   0
                                REG4:   0
                                REG5:   0
                                NUMO:   0                       ;CSR OFFSET FOR ERROR REPORT
                                NUM1:   0
                                RMTST:  0                       ; LONG TEST RAM FLAG
                                GDDAT:  0                       ;GOOD DATA
                                BDDAT:  0                       ;BAD DATA
                                BADLOC: 0                       ;FOR ERROR REPORT
                                TSTLOC: 0                       ;       "
                                BDLVL:  0                       ;       "
                                GDLVL:  0                       ;       "
                                GDVEC:  0                       ;       "
                                RATIO:  0                       ;CURRENT LINE SPEED RATE
                                BUF01:  .BLKW   16.             ;ERROR MESSAGE BUFFER
                                BUFLEN: .WORD   CSRLEN          ;BUFFER LENGTH INITED TO CSR LENGTH

                                ;*******************************************************************************
                                ;* REVISION CODE FOR TEST
                                ;*******************************************************************************

                                REVCOD: 3

                                ;*******************************************************************************
                                ;* POINTERS TO KMV11 VECTORS AND REGISTERS
                                ;*******************************************************************************

                                KMVCSR: 0                       ;CSR ADDRESS
                                KMVV00: 0                       ;POINTER TO INTERRUPT VECTOR XX0
                                KMVV04: 0                       ;POINTER TO INTERRUPT VECTOR XX4
                                KMVLVL: 0                       ;INTERRUPT SERVICING RELATIVE LEVEL


                                ;*******************************************************************************
                                ;* SPECIAL LOCATION RESERVED FOR TESTS ON MORE THAN ONE UNIT
                                ;*******************************************************************************

                                L$SW:   .WORD   0
                                LOCK:   .WORD   0               ; ADDRESS FOR LOCK CURRENT DATA
                                INIFLG: .BYTE   0               ;PROGRAM INITIALIZING FLAG
                                        .EVEN
                                LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
                                QV.FLG: .BYTE   0               ;QUICK VERIFY FLAG
                                        .EVEN
                                UUT:    .WORD   0               ;CURRENT UNIT UNDER TEST
                                UNIT:   .WORD   0               ;USED FOR UNIT NUMBER
                                ADDR:   .WORD   0
                                MAXERR: .WORD   5               ;MAX ERROR NUMBER ALLOWED
                                ERRCNT: .WORD   0               ;CURRENT ERROR NUMBER
                                NXMLOW: .WORD   160000          ; /V01.04/ FIRST USER SPACE I/O
                                NXMHIG: .WORD   277             ; ADDRESS FOR NXM MEMORY TEST
```

```
                                ;*******************************************************************************
                                ;* DATA STORAGE
                                ;*******************************************************************************

002532                          REGSPC: .BLKW   16.             ;GOOD IMAGE CSR

002572                          WRKSPC: .BLKW   16.             ;WORK SPACE

002632                          TRXBUF:                         ;TX BUFFER FOR DMA
        000004                          .REPT   4               ;[0,1,2,3...376,0,1,2...376....376]
                                        A=0
                                        .REPT   377
                                        .BYTE   A
                                        A=A+1
                                        .ENDR
                                        .ENDR
        001774                  TDMALG=.-TRXBUF

004626                          RCVBUF: .BLKB   1020.           ;RX BUFFER FOR DMA

006622  000020                  PNTBL:  .REPT   16.             ; /V01.04/ FIRST PASS TABLE
                                        .WORD   0
                                        .ENDR


                                ;       KMV ROM PART NUMBER PRINT OUT
                                .NLIST BEX
006662      045     116     045 MPNUML: .ASCIZ  /%N%A LOW  PROM PART NUMBER  : %T/
006723      045     116     045 MPNUMH: .ASCIZ  /%N%A HIGH PROM PART NUMBER  : %T/
006764                          PARTNB: .BLKB   36.             ;EPROM LOW AND HIGH PART NUMBER /V01.04/
007030      000     000                 .BYTE   0,0
                                .EVEN

                                ;TEST34 (MODEM SIGNALS TEST) MESSAGE
007032      045     116     045 EM0052: .ASCIZ  /%N%A TEST NOT RUN - NO LOOPBACK CONNECTOR INSERTED /

                                ; TEST37 (LONG RAM TEST) MESSAGE
007116      045     116     045 MTST37: .ASCII  /%N%A TEST37 STARTED: /
007143      045     116     045         .ASCIZ  /%N%A ITS DURATION IS ABOUT 1H15 /
                                .LIST BEX
                                .EVEN

                                ;*******************************************************************************
                                ;* PATTERN TABLE
                                ;*******************************************************************************

007204  000000                  PATTRN: 000000
007206  177777                          177777
007210  052525                          052525
007212  125252                          125252
007214  177776                          177776
007216  177775                          177775
007220  177773                          177773
007222  177767                          177767
007224  177757                          177757
007226  177737                          177737
```

```
                                                177677
                                                177577
                                                177377
                                                176777
                                                175777
                                                173777
                                                167777
                                                157777
                                                137777
                                                077777
                                                100000
                                                040000
                                                020000
                                                010000
                                                004000
                                                002000
                                                001000
                                                000400
                                                000200
                                                000100
                                                000040
                                                000020
                                                000010
                                                000004
                                                000002
                                                000001
                                                000000
000112                                  PATLGH=.-PATTRN
                                        ;
                                        ; THIS TABLE IS USED IN THE TEST10.
                                        ; EACH ENTRY CONTAINS ONE BYTE FOR THE "FROM" OFFSET AND ANOTHER FOR
                                        ; THE "TO" OFFSET.
                                        ;
004     002                     REGTAB: .BYTE   04,02
004     006                             .BYTE   04,06
006     010                             .BYTE   06,10
010     012                             .BYTE   10,12
012     014                             .BYTE   12,14
014     016                             .BYTE   14,16
016     020                             .BYTE   16,20
020     022                             .BYTE   20,22
022     024                             .BYTE   22,24
024     026                             .BYTE   24,26
026     030                             .BYTE   26,30
030     032                             .BYTE   30,32
032     034                             .BYTE   32,34
034     036                             .BYTE   34,36
036     004                             .BYTE   36,04
000000                                  .WORD   0


                                        ;
                                        ;       LINE CLOCK RATE RATIO IN BCD (4 DECADES)
                                        ;

007356                          RATIOT: ; DIVIDER VALUE IN BCD TO BE LOADED IN LINE COUNTER

024200                                  24200                           ;2.4K RATE=0
```

```
489 007360  003440                         3440                            ;9,6K RATE=1
490 007362  001540                         1540                            ;19.2KRATE=2
491 007364  000504                         504                             ;48K  RATE=3
492 007366  000444                         444                             ;56K  RATE=4
493 007370  000410                         410                             ;64K  RATE=5
494 007372  000226                         226                             ;72K  RATE=6
495 007374  000000                         0
496                                ;        151                             ;100K
497                                ;*************************************************************************
498                                ;* STACK USED FOR SUBROUTINE LINKAGE
499                                ;*************************************************************************
500
501 007376                                 .BLKW   100
502 007576                         SSTACK:
503
504
505
506
507
508
509
```

```
                        .SBTTL  GLOBAL TEXT SECTION

                        ;*****************************************************************************
                        ;*      THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                        ;*      MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                        ;*      MORE THAN ONE TEST.
                        ;*****************************************************************************

                        ;;*************************************************************************
                        ;* NAMES OF DEVICES SUPPORTED BY PROGRAM
                        ;;*************************************************************************
007576                          DEVTYP  <M7502>


                        .IF NE HELP
                        ;*****************************************************************************
                        ;     INSERT THE FORMAT STATEMENTS USED IN THE VARIOUS PRINT CALLS.
                        ;     USE THE .ASCIZ STATEMENT.
                        ;*****************************************************************************
                        .ENDC
```

```
538                             .SBTTL  GLOBAL SUBROUTINES
539
540
541
542
543                             ;////////////////////////////////////////////////////////////////////////////////
544                             ;/      THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
545                             ;////////////////////////////////////////////////////////////////////////////////
546                             .IF NE HELP
547
548
549                             ; SUBORDINATE ROUTINES USED:
550
551                             ;       LIST THE SUBROUTINES CALLED BY THIS SUBROUTINE.
552                             ;********************************************************************************
553
554                             ; FUNCTIONAL SIDE EFFECTS:
555
556                             ;********************************************************************************
557                             ;       DESCRIBE ANY EFFECTS THIS SUBROUTINE MAY HAVE UPON OTHER
558                             ;       MODULES OF THE DIAGNOSTIC PROGRAM.  AN EXAMPLE OF THIS IS
559                             ;       THE SUBROUTINE INHIBITS ALL INTERRUPTS WITH PRIORITY 7.
560                             ;********************************************************************************
561
562                             ; CALLING SEQUENCE:
563
564                             ;********************************************************************************
565                             ;       GIVE THE EXACT CALLING SEQUENCE USED TO ACCESS THIS SUBROUTINE.
566                             ;       FOR EXAMPLE:    MOV COUNT,R1    ;MOVE INPUT TO R1
567                             ;                       JSR     PC,ROUTINE      ;GO TO ROUTINE
568                             ;                       BCS     ERROR           ;CARRY SET IF ROUTINE HAD ERROR
569                             ;********************************************************************************
570                             ;--
571
572                             ;********************************************************************************
573                             ;     INSERT THE CODE FOR THIS SUBROUTINE.   THE NAME OF THE SUBROUTINE SHOULD
574                             ;     BE DEFINED WITH A DOUBLE-COLON (::);  THIS WILL MAKE THE SUBROUTINE GLOBAL.
575                             ;********************************************************************************
576                             .ENDC
577
578                             .IF NE HELP
579                             ;********************************************************************************
580                             ;     BEGIN EACH SUBROUTINE AT THE TOP OF A NEW PAGE.
581                             ;********************************************************************************
582
583                             ;++
584                             ; FUNCTIONAL DESCRIPTION:
585                             ;    SUBROUTINE TO....
586                             ;--
587
588                             ;********************************************************************************
589                             ;       COMPLETE THE "SUBROUTINE TO...." STATEMENT WITH A FUNCTIONAL
590                             ;       DESCRIPTION OF THIS SUBROUTINE.
591                             ;********************************************************************************
592
593                             ; INPUTS:
594
```

```
595                     ;*****************************************************************************
596                     ;       LIST THE INPUT DATA THAT ARE EXPLICITLY PASSED TO THIS SUBROUTINE.
597                     ;*****************************************************************************
598
599                     ; IMPLICIT INPUTS:
600
601                     ;*****************************************************************************
602                     ;       LIST THE INPUT DATA THAT ARE IMPLICITLY USED BY THIS SUBROUTINE;
603                     ;       FOR EXAMPLE, DATA READ FROM COMMON AREAS.
604                     ;*****************************************************************************
605
606                     ; OUTPUTS:
607
608                     ;*****************************************************************************
609                     ;       LIST THE OUTPUT DATA THAT ARE EXPLICITLY GIVEN BY THIS SUBROUTINE
610                     ;*****************************************************************************
611
612                     ; IMPLICIT OUTPUTS:
613
614                     ;*****************************************************************************
615                     ;       LIST THE OUTPUT DATA THAT ARE IMPLICITLY GIVEN BY THIS SUBROUTINE;
616                     ;       FOR EXAMPLE, DATA STORED IN COMMON AREAS.
617                     ;*****************************************************************************
618                     .ENDC
619
620
621
622
623
624
625
626                     ;*****************************************************************************
627                     ;* LIST OF THE COMMON MACRO AND SUBROUTINES
628                     ;*
629                     ;* -TO CHECK MAXIMUM ERRORS IS NOT REACHED       : CALL  CHKMAX
630                     ;* -TO REPEAT N TIMES A FUNCTION                 : DO     <FUNCTION>,N
631                     ;* -TO WAIT TILL SELO/INTFLG OR TIME-OUT         : WAIT   O(1 OR 2),N.
632                     ;* -TO DISPATCH ACCORDING TO ERROR REPORT        : DSPACH N.
633                     ;* -TO SAVE GENERAL REGISTERS                    : SAVE   012...
634                     ;* -TO RESTORE GENERAL REGISTERS                 : GET    012...
635                     ;* -TO COMPARE BLOCK OF DATA (BY WORD)           : CALL COMPAR
636                     ;* -TO COMPARE BLOCK OF DATA (BY BYTE)           : CALL COMPRB
637                     ;* -TO CLEAR A BUFFER                            : CALL CLEAR
638                     ;* -TO COPY ONE BUFFER IN A SECOND ONE           : CALL COPY
639                     ;* -TO CLEAR ALL CSR'S                           : CALL CLRKMV
640                     ;* -TO SET MAINTENANCE MODE                      : MODE   2(3)
641                     ;* -TO ACTIVATE DCT11 TEST ROUTINE               : TESTNB N.
642                     ;* -TO MAKE A DELAY OF N*100 MICROSECONDS        : DELAY  N.
643                     ;*
644                     ;*****************************************************************************
```

```
                                        ;++
                                        ; ROUTINE TO CHECK THAT THE MAXIMUM NUMBER OF ERRORS IS NOT REACHED
                                        ;
                                        ; OUTPUT: RETURN TO CALLER IF MAX NUMBER IS NOT REACHED
                                        ;         OTHERWISE PRINT ERROR MESSAGE  AND DROP UNIT
                                        ;--
007604                                  CHKMAX: INLOOP                  ;LOOPING ON ERROR ?
007606                                          BCOMPLETE 1$            ;YES, EXIT
007610                                          RFLAGS  R0              ;GET OPERATORS FLAG
007612  032700  000040                          BIT     #IDU,R0         ;IS DROPPING INHIBITTED ?
007616  001026                                  BNE     1$              ;YES, EXIT
007620  005237  002524                          INC     ERRCNT          ;INC COUNTER
007624  023737  002522  002524                  CMP     MAXERR,ERRCNT   ;MAX NUMBER REACHED ?
007632  002020                                  BGE     1$              ;NO, RETURN TO CALLER

007634                                          PRINTF  #NERRS,MAXERR,UUT       ;PRINT CORRESPONDING MESSAGE
007664                                          DODU    UUT             ;DROP UNIT
007672                                          DOCLN                   ;END SUBPASS
007674                                  1$:
007674  000207                                  RETURN

007676     045     116     045  NERRS:  .ASCIZ  /%N%AMORE THAN %D3%A  ERRORS ON UNIT %D2/
007701     101     115     117
007704     122     105     040
007707     124     110     101
007712     116     040     045
007715     104     063     045
007720     101     040     040
007723     105     122     122
007726     117     122     123
007731     040     117     116
007734     040     125     116
007737     111     124     040
007742     045     104     062
007745     000

                                        .EVEN
```

```
                        ;++
                        ; ROUTINE TO SAVE GENERAL REGISTERS
                        ;
                        ; DESCRIPTION: PERMITS TO SAVE GENERAL REGITERS R0 TO R7
                        ;
                        ; CALLING SEQUENCE: SAVE        123...
                        ;
                        ; INPUTS: REGISTER NUMEROS LIST
                        ;
                        ; OUTPUTS: REG(N)=R(N)
                        ;
                        ; CAUTIONS: NONE
                        ;--


                              .MACRO  SAVE    ARGLST
                                      .IRPC   N,<ARGLST>
                                      MOV     R'N,REG'N
                                      .ENDR
                              .ENDM
```

```
693
694                                     ;++
695                                     ; ROUTINE TO RESTORE GENERAL REGISTERS
696                                     ;
697                                     ; DESCRIPTION: PERMITS TO RESTORE GENERAL REGISTERS R0 TO R7
698                                     ;
699                                     ; CALLING SEQUENCE: GET      123....
700                                     ;
701                                     ; INPUTS: REGISTER NUMEROUS LIST
702                                     ;
703                                     ; OUTPUTS: NONE
704                                     ;
705                                     ; CAUTIONS: NONE
706                                     ;--
707
708
709
710                                             .MACRO  GET     ARGLST
711                                                     .IRPC   N,<ARGLST>
712                                                     MOV     REG'N,R'N
713                                                     .ENDR
714                                             .ENDM
715
```

```
                        ;++
                        ; ROUTINE TO DISPATCH ACCORDING TO ERROR REPORT
                        ;
                        ; DESCRIPTION: DATA IN SEL2 IS USED AS AN OFFSET TO POINT TO AN ERROR BRANCH
                        ;              TABLE WHICH FOLLOWS THE CALL NON-EXPECTED OFFSET IN SEL2
                        ;              POINTS TO THE ADDRESS FOLLOWING THE LAST BRANCH.
                        ;
                        ; CALLING SEQUENCE: DSPACH      N.
                        ;                   BR  3$
                        ;                   BR  4$
                        ;                   BR  5$
                        ;                   .
                        ;                   .
                        ;                   .
                        ;
                        ; INPUTS: N.=DECIMAL NUMBER OF BRANCH
                        ;         SEL2=POINT TO ERROR OFFSET
                        ;
                        ; OUTPUTS: BAD=CONTENTS OF SEL2
                        ;          NUMO=2
                        ;
                        ; CAUTIONS: NONE
                        ;--


                                .MACRO  DSPACH  N
                                        JSR     R5,TSTSL2
                                        N
                                .ENDM


007746                          TSTSL2: SAVE    12
007756  012501                          MOV     (R5)+,R1                ;GET NUMBER OF BRANCH
007760  006301                          ASL     R1                      ;*2
007762  013702  002474                   MOV     KMVCSR,R2               ;GET CSR ADDRESS
007766  016237  000002  002352           MOV     SEL2(R2),BAD            ;GET SEL2 CONTENTS
007774  005737  002352                   TST     BAD                     ;IS THERE ANY OFFSET?
010000  001416                          BEQ     1$                      ;IF NOT
010002  032737  000001  002352           BIT     #BIT0,BAD               ;EVEN OFFSET?
010010  001012                          BNE     1$                      ;IF ODD
010012  023701  002352                   CMP     BAD,R1                  ;CURRENT OFFSET <= TO MAXI OFFSET?
010016  101007                          BHI     1$                      ;IF NOT
010020  005337  002352                   DEC     BAD                     ;ADJUST OFFSET
010024  005337  002352                   DEC     BAD
010030  063705  002352                   ADD     BAD,R5                  ;POINT TO CORRESPONDING BRANCH
010034  000410                          BR      2$

010036  012737  000002  002402   1$:     MOV     #2,NUMO                 ;GET CSR OFFSET
010044                                  ERRHRD  13,EM0012,PRBAD         ;REPORT ERROR IF BAD SEL2
010054  060105                          ADD     R1,R5
010056                           2$:
010056                                  GET     12
010066  000205                          RTS     R5
```

```
                        ;**
                        ; ROUTINE TO WAIT TILL SELO/INTFLG OR A TIME-OUT
                        ;
                        ; DESCRIPTION: WHILE WE ARE WAITING FOR N*1MS, A TEST IS DONE FOR AN EVENT
                        ;              SUCH AS A END OF DCT11 TEST OR AN INTERRUPT. THE FORMER IS
                        ;              DONE BY POLLING SELO FOR 0 OR 100. THE LATTER IS DONE BY
                        ;              POLLING INTFLG .IF TIME ELAPSED BEFORE ANY EVENT, TIMOUT FLAG
                        ;              IS SET.POSSIBLE EXITS:
                        ;               -CASE OF SELO   PC IF SELO=0
                        ;                               PC+2 IF SELO=100
                        ;                               PC+4 IF TIME-OUT
                        ;               -CASE IF INT.   PC IF INTFLG SET
                        ;                               PC+2 IF TIME-OUT
                        ;               -CASE TIME-OUT  PC
                        ;
                        ; CALLING SEQUENCE: WAIT        0(1 OR 2),N,M
                        ;
                        ; INPUTS: FIRST CALL PARAMETER=0 IN CASE OF INTFLG
                        ;                               =1 IN CASE OF SELO
                        ;                               =2 IN CASE OF A SIMPLE WAIT
                        ;         N=DECIMAL NUMBER OF MS
                        ;         M=MULTIPLICATOR FACTOR FOR N (ONLY USED IN WAIT1)
                        ;
                        ; OUTPUTS:
                        ;          NUM0=0
                        ;          BAD=CONTENTS OF SELO
                        ;
                        ; CAUTIONS: NONE
                        ;--


                                .MACRO  WAIT    TYP,N,M
                                        JSR     R5,WAIT'TYP
                                        N
                                        .IF NB M
                                        M
                                        .IFF
                                        1
                                        .ENDC
                                .ENDM


                        WAIT0:  SAVE    1
010070                          MOV     (R5)+,R1                ;GET TIME LENGTH
010074  012501                  TST     (R5)+
010076  005725
                        1$:     DELAY   10.                     ;WAIT FOR 1MS
010100
010130  005737  002356          TST     INTFLG                  ;ANY INTERRUPT?
010134  001004                  BNE     2$                      ;IF YES
010136                          BREAK
010140  077121                  SOB     R1,1$                   ;LOOP TILL TIME ELAPSES
010142  062705  000002          ADD     #2,R5                   ;PC+2
010146                  2$:     GET     1
010152  000205                  RTS     R5
```

```
831
832 010154                              WAIT1:  SAVE    12
833 010164  012501                              MOV     (R5)+,R1        ;GET TIME LENGTH
834 010166  012502                              MOV     (R5)+,R2
835 010170                              1$:     DELAY   10.             ;WAIT FOR 1MS
836 010220  017737  172250  002352              MOV     @KMVCSR,BAD     ;READ SEL0
837 010226  005737  002352                      TST     BAD             ;SEL0=0?
838 010232  001415                              BEQ     2$              ;IF YES
839 010234  022737  000100  002352              CMP     #100,BAD        ;SEL0=100?
840 010242  001407                              BEQ     3$              ;IF YES
841 010244                                      BREAK
842 010246  077130                              SOB     R1,1$           ;LOOP TILL TIME-OUT
843 010250  016501  177774                      MOV     -4(R5),R1       ;REINIT R1
844 010254  077233                              SOB     R2,1$           ;LOOP UNTIL R2 REACHES 0
845 010256  062705  000002                      ADD     #2,R5           ;PC+2
846 010262  062705  000002              3$:     ADD     #2,R5           ;PC+2
847 010266  005037  002402              2$:     CLR     NUM0            ;NUM0=0
848 010272                                      GET     12
849 010302  000205                              RTS     R5
850
851 010304                              WAIT2:  SAVE    1
852 010310  012501                              MOV     (R5)+,R1        ;GET TIME LENGTH
853 010312  005725                              TST     (R5)+
854 010314                              1$:     DELAY   10.             ;WAIT FOR 1MS
855 010344                                      BREAK
856 010346  077116                              SOB     R1,1$           ;LOOP TILL TIME ELAPSES
857 010350                              2$:     GET     1
858 010354  000205                              RTS     R5
```

```
                                ;**
                                ; ROUTINE TO COMPARE BLOCKS OF DATA
                                ;
                                ; DESCRIPTION: TWO BUFFERS ARE COMPARED WORD BY WORD.
                                ;
                                ;
                                ; INPUTS: R1 SOURCE BUFFER 1 ADDRESS
                                ;         R2 SOURCE BUFFER 2 ADDRESS
                                ;         BUFLEN CONTAINS THE LENGTH OF BUFFER TO COMPARE (IN GENERAL = CSRLEN)
                                ;
                                ; OUTPUTS: R1,R2 DESTROYED
                                ;          GOOD  CONTAINS THE BUFFER 1 WORD
                                ;          BAD   CONTAINS THE BUFFER 2 WORD
                                ;          CARRY SET IF A MISMATCH OCCURS
                                ;
                                ;--
010356                          COMPAR: SAVE    34
010366  013703  002470                  MOV     BUFLEN,R3               ;GET NUMBER OF WORDS
010372  005004                          CLR     R4                      ;CURRENT OFFSET=0
010374  012137  002350          1$:     MOV     (R1)+,GOOD              ;READ SOURCE
010400  012237  002352                  MOV     (R2)+,BAD               ;READ DESTINATION
010404  023737  002350  002352          CMP     GOOD,BAD                ;THE SAME?
010412  001010                          BNE     3$                      ;IF NOT
010414  005724                          TST     (R4)+                   ;ADD 2 TO OFFSET
010416  077312                          SOB     R3,1$                   ;DEC R3 AND CONTINUE IF NOT NULL
010420  000241                          CLC
010422                          2$:     GET     34
010432  000207                          RETURN
010434  010437  002402          3$:     MOV     R4,NUMO                 ;SAVE CURRENT OFFSET
010440  000261                          SEC
010442  000767                          BR      2$
```

```
                                ;++
                                ; ROUTINE TO COMPARE BLOCKS OF DATA
                                ;
                                ; DESCRIPTION: TWO BUFFERS ARE COMPARED BYTE BY BYTE.
                                ;
                                ;
                                ; INPUTS: R1 SOURCE BUFFER 1 ADDRESS
                                ;         R2 SOURCE BUFFER 2 ADDRESS
                                ;         BUFLEN CONTAINS THE LENGTH OF BUFFER TO COMPARE
                                ;
                                ; OUTPUTS: R1,R2 DESTROYED
                                ;          GOOD  CONTAINS THE BUFFER 1 BYTE
                                ;          BAD   CONTAINS THE BUFFER 2 BYTE
                                ;          CARRY SET IF A MISMATCH OCCURS
                                ;
                                ;--
010444                          COMPRB: SAVE    3
010450  013703  002470                  MOV     BUFLEN,R3               ;GET NUMBER OF BYTES
010454  005037  002350                  CLR     GOOD                    ;CLEAN LOCATION
010460  005037  002352                  CLR     BAD                     ;CLEAN LOCATION
010464  112137  002350          1$:     MOVB    (R1)+,GOOD              ;READ SOURCE
010470  112237  002352                  MOVB    (R2)+,BAD               ;READ DESTINATION
010474  123737  002350  002352          CMPB    GOOD,BAD                ;THE SAME?
010502  001005                          BNE     3$                      ;IF NOT
010504  077311                          SOB     R3,1$                   ;DEC R3 AND CONTINUE IF NOT NULL
010506  000241                          CLC
010510                          2$:     GET     3
010514  000207                          RETURN
010516  000261                  3$:     SEC
010520  000773                          BR      2$
```

```
                        ;++
                        ; ROUTINE TO SET MAINTENANCE MODE
                        ;
                        ; DESCRIPTION: SET MAINTENANCE MODE 2 OR 3 AND CHECK THAT MASTER CLEAR IS
                        ;              RESET BY THE DCT11. ERROR REPORT IF NOT.
                        ;              IN MODE 3 THE DCT11 LOOPS ON ITSELF WHILE IN MODE 2 THE
                        ;              DCT11 IS READY TO RUN A TEST
                        ;
                        ; CALLING SEQUENCE: MODE        N
                        ;
                        ; INPUTS: N=2 FOR MODE 2 OR =3 FOR MODE 3
                        ;
                        ; OUTPUTS: NONE
                        ;
                        ; CAUTIONS: NONE
                        ;--


                                .MACRO  MODE    N
                                        JSR     PC,MAINM'N
                                .ENDM


010522  005077  171746          MAINM3: CLR     @KMVCSR                 ;CLEAN CSR
010526  012777  044000  171740          MOV     #MCLR!MAINT2,@KMVCSR    ;SET MODE 3
010534                                  DELAY   1                       ;WAIT FOR 100MICROS
010564  022777  004000  171702          CMP     #MAINT2,@KMVCSR         ;MASTER CLEAR RESET ?
010572  001404                          BEQ     1$                      ;YES, CORRECT
010574  012737  004000  002350          MOV     #MAINT2,GOOD            ;LOAD GOOD VALUE FOR ERROR MESSAGE
010602  000401                          BR      MAIERR                  ;JOIN COMMON ERROR CODE
010604  000207                  1$:     RTS     PC                      ;RETURN
010606  005037  002402          MAIERR: CLR     NUM0                    ;OFFSET IS 0
010612  017737  171656  002352          MOV     @KMVCSR,BAD             ;LOAD BAD VALUE
010620                                  ERRHRD  2,EM0002,PRSEL          ;ERROR REPORT IF NOT
010630  000261                          SEC
010632  000207                          RETURN


010634  005077  171634          MAINM2: CLR     @KMVCSR                 ;CLEAN CSR
010640  012777  041000  171626          MOV     #MCLR!MAINT1,@KMVCSR    ;SET MODE 2
010646                                  DELAY   2                       ;WAIT FOR 200MICROS
010676  032777  040000  171570          BIT     #MCLR,@KMVCSR           ;MASTER CLEAR RESET ?
010704  001431                          BEQ     1$                      ;YES CORRECT
010706  013705  002474                  MOV     KMVCSR,R5               ;/V01.01/ GET CSR ADDRESS IN R5
010712  122765  000100  000000          CMPB    #100,SEL0(R5)           ;/V01.01/ IS IT A STACK PROBLEM ?
010720  001017                          BNE     1000$                   ;/V01.01/ NO, GENERAL ERROR
010722  016537  000002  002352          MOV     SEL2(R5),BAD            ;/V01.01/ GET BAD VALUE
010730  016537  000004  002350          MOV     SEL4(R5),GOOD           ;/V01.01/ GET EXPECTED ONE
010736  016537  000006  002354          MOV     SEL6(R5),SPADDR         ;/V01.01/ LOAD STACK LOCATION
010744                                  ERRHRD  0,EM0000,PRSTAK         ;/V01.01/ STATCK ERROR MESSAGE
010754  000261                          SEC
010756  000207                          RETURN
010760  012737  001000  002350  1000$:  MOV     #MAINT1,GOOD            ;LOAD GOOD VALUE FOR ERROR MESSAGE
010766  000707                          BR      MAIERR                  ;JOIGN COMMON ERROR CODE
```

```
    982 010770  000207                  1$:     RTS     PC
    983
    984
```

```
                        ;++
                        ; ROUTINE TO SELECT DCT11 TEST ROUTINE
                        ;
                        ; DESCRIPTION: LOAD THE TEST NUMBER IN BSELO AND SET TSTNUM BIT TO
                        ; VALIDATE THE NUMBER IN BSELO. BIT 7 IN SEL0 IS ALSO SET IF THE
                        ; INTERNAL LOOPBACK IS SELECTED.
                        ;
                        ; CALLING SEQUENCE: TESTNB      N
                        ;
                        ; INPUTS: N=TEST NUMBER IN OCTAL
                        ;
                        ; OUTPUTS: NONE
                        ;
                        ; CAUTIONS: NONE
                        ;--


                                .MACRO  TESTNB  N
                                        JSR     R5,TSTNB
                                        N
                                .ENDM


010772  042715  177600  TSTNB:  BIC     #177600,(R5)            ;KEEP ONLY VALID BITS
010776  052715  010000          BIS     #TSTNUM,(R5)            ;SET THIS BIT TO BE RECOGNIZED BY THE FW
011002  005737  002366          TST     LCLOOP                  ;LOOPBACK CONNECTOR INSTALLED ?
011006  001002                  BNE     1$                      ;YES, NO INTERNAL LOOP SELECTED
011010  152715  000200          BISB    #200,(R5)               ;SET BIT7 IN SELO TO SELECT INTERNAL LOOP
011014  012577  171454  1$:     MOV     (R5)+,@KMVCSR           ;LOAD NUMBER IN CSR
011020                          DELAY   10.                     ;WAIT FOR 1MS
011050  000205                  RTS     R5
```

```
1022
1023                                    ;++
1024                                    ; ROUTINE TO CLEAR A SPECIFIED BUFFER
1025                                    ;
1026                                    ;
1027                                    ; INPUTS: R1 BUFFER ADDRESS
1028                                    ;         BUFLEN CONTAINS THE LENGTH
1029                                    ;
1030                                    ; OUTPUTS: R1 DESTROYED
1031                                    ;
1032                                    ;--
1033 011052                            CLEAR:  SAVE    3
1034 011056  013703  002470                    MOV     BUFLEN,R3               ;GET NUMBER OF WORDS
1035 011062  005021                     1$:     CLR     (R1)+                   ;CLEAR BUFFER
1036 011064  077302                             SOB     R3,1$                   ;DEC COUNTER AND CONTINUE IF NOT NULL
1037 011066                                     GET     3
1038 011072  000207                             RETURN
```

```
1040
1041                                    ;++
1042                                    ; ROUTINE TO COPY ONE BUFFER IN A SECOND ONE
1043                                    ; THE BUFFER LENGTH IS EQUAL TO CSRLEN
1044                                    ;
1045                                    ; INPUTS: R1 SOURCE BUFFER ADDRESS
1046                                    ;         R2 DESTINATION BUFFER ADDRESS
1047                                    ;
1048                                    ; OUTPUTS: R1,R2 DESTROYED
1049                                    ;
1050                                    ;--
1051 011074                            COPY:   SAVE    3
1052 011100  012703  000020                    MOV     @CSRLEN,R3              ;GET NUMBER OF WORDS
1053 011104  012122                    1$:     MOV     (R1)+,(R2)+             ;COPY FROM SOURCE TO DESTINATION BUFFER
1054 011106  077302                            SOB     R3,1$                   ;DEC COUNTER AND CONTINUE IF NOT NULL
1055 011110                                    GET     3
1056 011114  000207                            RETURN
1057
```

```
1059
1060                                    ;++
1061                                    ; ROUTINE TO CLEAR KMV11 CONTROL AND STATUS REGISTERS
1062                                    ;
1063                                    ; INPUTS: NONE
1064                                    ;
1065                                    ; OUTPUTS: R1 DESTROYED
1066                                    ;          R5 POINTS ON CSR
1067                                    ;          BUFLEN CONTAINS CSRLEN
1068                                    ;
1069                                    ;--
1070 011116  013701  002474            CLRKMV: MOV     KMVCSR,R1               ;GET CSR ADDRESS
1071 011122  012737  000020  002470            MOV     @CSRLEN,BUFLEN          ;LOAD NB OF WORDS TO CLEAR
1072 011130  004737  011052                    CALL    CLEAR
1073 011134  012705  000340                    MOV     @PRI07,R5               ; SET HOST PRIORITY /V01.04/
1074 011140  106405                            MTPS    R5
1075 011142  013705  002474                    MOV     KMVCSR,R5               ;GET CSR ADDRESS IN R5
1076 011146  000207                            RETURN
```

```
1078
1079                                    ;++
1080                                    ; ROUTINE TO REPEAT N TIMES A FUNCTION
1081                                    ;
1082                                    ; DESCRIPTION: NONE
1083                                    ;
1084                                    ; CALLING SEQUENCE: DO  <ASL    R1>,10.
1085                                    ;                       REPEAT 10 TIMES THE LINE CODE:ASL   R1
1086                                    ;
1087                                    ; INPUTS: THE LINE CODE TO REPEAT BETWEEN <..>
1088                                    ;         THE DECIMAL NUMBER FOR REPEAT
1089                                    ;
1090                                    ; OUTPUTS: NONE
1091                                    ;
1092                                    ; CAUTIONS: NONE
1093                                    ;--
1094
1095
1096
1097                                            .MACRO  DO      FUNCTION,N
1098                                                    .REPT   N
1099                                                    FUNCTION
1100                                                    .ENDR
1101                                            .ENDM
```

```
1103
1104
1105                            ;++
1106                            ; ROUTINE TO PRINT TEST NUMBER
1107                            ;
1108                            ; DESCRIPTION: NONE
1109                            ;
1110                            ; CALLING SEQUENCE: BADHEAD
1111                            ;
1112                            ; INPUTS: L$TEST=TEST NUMBER
1113                            ;
1114                            ; OUTPUTS: NONE
1115                            ;
1116                            ; CAUTION: NONE
1117                            ;--
1118
1119
1120
1121                                    .MACRO  ED$CALL XY
1122                                            .LIST
1123                            ;++
1124                            ;
1125                            ;
1126                            ;
1127                            ;
1128                            ;                       ** TEST'XY' **
1129                            ;
1130                            ;
1131                            ;
1132                            ;-
1133                                            .NLIST
1134                                    .ENDM
1135
1136
1137
1138                                    .MACRO  BADHEAD
1139                                            .RADIX  10
1140                                            ED$CALL \T$TESTNUM+1
1141                                            .RADIX  8
1142                                    .ENDM
1143
1144
```

```
1146                                    .SBTTL  GLOBAL ERROR REPORT SECTION
1147
1148                                    ;////////////////////////////////////////////////////////////////////////////////
1149                                    ;/      THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
1150                                    ;/      THAT ARE USED IN MORE THAN ONE TEST.
1151                                    ;////////////////////////////////////////////////////////////////////////////////
1152                                    .NLIST BEX
1153
1154 011150     040     040     102    TIM:    .ASCIZ  /  BUS TIMEOUT/
1155 011166     040     123     124    EM0000: .ASCIZ  / STACK PROBLEM WHEN MODE 2 SELECTED/
1156 011232     040     124     111    EM0001: .ASCIZ  / TIME-OUT TRAP ON REFERENCING Q-BUS DEVICE REGISTERS/
1157 011317     040     115     101    EM0002: .ASCIZ  / MASTER CLEAR NOT RESET BY THE KMV11/
1158 011364     040     113     115    EM0003: .ASCIZ  / KMV11 CONTROL AND STATUS REGISTERS CANNOT BE CLEARED/
1159 011452     040     104     101    EM0004: .ASCIZ  / DATA COMPARE ERROR ON CSR'S Q-BUS ACCESS/
1160 011524     040     124     110    EM0005: .ASCIZ  / THE KMV11 CANNOT CORRECTLY ACCESS SEL0 OR ENTER MODE 2/
1161 011614     040     113     115    EM0006: .ASCIZ  / KMV11 NATIVE FIRMWARE CHECKSUM ERROR/
1162 011662     040     122     105    EM0007: .ASCIZ  / REVISION CODE DOES NOT MATCH/
1163 011720     040     113     115    EM0010: .ASCIZ  / KMV11 CAN'T CORRECTLY READ SEL2/
1164 011761     040     116     117    EM0011: .ASCIZ  / NO KMV11-C RESPONSE/
1165 012006     040     121     055    EM0012: .ASCIZ  / Q-BUS CAN'T CORRECTLY ACCESS ALL CSR'S/
1166 012056     040     122     101    EM0013: .ASCIZ  / RAM LOCATION CANNOT BE CLEARED/
1167 012116     040     104     101    EM0014: .ASCIZ  / DATA COMPARE ERROR ON SELECTED RAM LOCATION/
1168 012173     040     104     101    EM0015: .ASCIZ  / DATA COMPARE ERROR ON RAM AT LOCATION NOT UNDER TEST/
1169 012261     040     104     101    EM0016: .ASCIZ  / DATA COMPARE ERROR ON KMV11 INTERNAL R-W REGISTERS/
1170 012345     040     121     055    EM0017: .ASCIZ  / Q-BUS INTERRUPT AT AN UNEXPECTED PRIORITY LEVEL/
1171 012426     040     116     117    EM0020: .ASCIZ  / NO INTERRUPT OCCURS ON Q-BUS/
1172 012464     040     116     117    EM0021: .ASCIZ  / NO KMV11 INTERRUPT WHEN CSR'S WERE ACCESSED /
1173 012542     040     125     116    EM0022: .ASCIZ  / UNEXPECTED KMV11 INTERRUPT DURING A CSR ACCESS/
1174 012622     040     113     115    EM0023: .ASCIZ  / KMV11 INTERRUPTED AT ILLEGAL VECTOR WHEN Q-BUS ACCESS CSR'S/
1175 012717     040     122     105    EM0024: .ASCIZ  / REAL TIME CLOCK INTERRUPT OCCURS TOO EARLY/
1176 012773     040     122     105    EM0025: .ASCIZ  / REAL TIME CLOCK COUNTER RUNS BUT NO INTERRUPT OCCURS/
1177 013061     040     122     105    EM0026: .ASCIZ  / REAL TIME CLOCK COUNTER DOES NOT RUN/
1178 013127     040     122     105    EM0027: .ASCIZ  / REAL TIME CLOCK CAUSES A KMV11 INTERRUPT AT ILLEGAL VECTOR/
1179 013223     040     113     115    EM0030: .ASCIZ  / KMV11 INTERRUPTED WHILE RTC DISABLED/
1180 013271     040     114     111    EM0031: .ASCIZ  / LINE CLOCK DOES NOT RUN CORRECTLY/
1181 013334     040     116     117    EM0032: .ASCIZ  / NO KMV11 INTERRUPT WHILE TRANSMIT INTERRUPT IS EXPECTED /
1182 013426     040     124     122    EM0033: .ASCIZ  / TRANSMIT INTERRUPT AT ILLEGAL VECTOR/
1183 013474     040     125     116    EM0034: .ASCIZ  / UNDERRUN INTERRUPT OCCURS/
1184 013527     040     113     115    EM0037: .ASCIZ  / KMV11 INTERRUPTED WHILE TRANSMIT INTERRUPT WAS DISABLED/
1185 013620     113     115     126    EM0040: .ASCIZ  /KMV11 NOT INTERRUPTED WHILE WAITING AN UNDERRUN/
1186 013700     125     116     104    EM0041: .ASCIZ  /UNDERRUN INTERRUPT AT ILLEGAL VECTOR/
1187 013745     113     115     126    EM0042: .ASCIZ  /KMV11 INTERRUPTED AGAIN AFTER ACKNOWLEDGE OF UNDERRUN INTERRUPT/
1188 014045     102     123     105    EM0043: .ASCIZ  /BSEL2 CAUSES AN UNEXPECTED INTERRUPT ON KMV11/
1189 014123     124     122     101    EM0044: .ASCIZ  /TRANSMISSION-RECEPTION PROCESSING FAILS - SEE EVENTS /
1190 014211     125     116     105    EM0045: .ASCIZ  /UNEXPECTED STATUS RECEIVED/
1191 014244     125     116     105    EM0046: .ASCIZ  /UNEXPECTED DATA RECEIVED/
1192 014275     122     105     103    EM0047: .ASCIZ  /RECEIVED INTERRUPT WHILE INTERRUPTS ARE DISABLED/
1193 014356     130     115     124    EM0050: .ASCIZ  /XMT-RCV TEST, KMV11 INTERRUPTED ON ILLEGAL VECTOR/
1194 014440     115     117     104    EM0051: .ASCIZ  /MODEM SIGNAL BIT SETTING NOT CORRECT/
1195 014505     125     116     105    EM0054: .ASCIZ  /UNEXPECTED END OF RECEIVE DMA INTERRUPT/
1196 014555     105     116     104    EM0055: .ASCIZ  /END OF TRANSMIT DMA WITHOUT BIT "BCNTIN" SET/
1197 014632     124     111     115    EM0056: .ASCIZ  /TIMEOUT DURING TRANSMIT DMA/
1198 014666     104     101     124    EM0057: .ASCIZ  /DATA COMPARE ERROR DURING TRANSMIT DMA/
1199 014735     104     101     124    EM0060: .ASCIZ  /DATA COMPARE ERROR DURING RECEIVE  DMA/
1200 015004     125     116     105    EM0061: .ASCIZ  /UNEXPECTED TIMEOUT DURING TRANSMIT DMA/
1201 015053     102     131     124    EM0062: .ASCIZ  /BYTE COUNT OVERFLOW DURING RECEIVE DMA/
1202 015122     124     111     115    EM0063: .ASCIZ  /TIMEOUT DURING RECEIVE DMA/
```

```
1203 015155     104     101     124  EM0064: .ASCIZ  /DATA COMPARE ERROR DURING DMA TRANSFER IN BOTH DIRECTIONS/
1204 015247     106     101     111  EM0065: .ASCIZ  /FAILURE TO DETECT A NON EXISTENT MEMORY DURING A DMA TRANSFER/
1205                                      .EVEN
```

```
1207                                    ;
1208                                    ;//////////////////////////////////////////////////////////////////////////////////////////
1209                                    ;ERROR REPORT SECTION
1210                                    ;//////////////////////////////////////////////////////////////////////////////////////////
1211                                    ;++
1212                                    ; ERROR REPORT FOR CSR ADDRESS ERROR
1213                                    ;--
1214 015346                            BGNMSG  PADFLT
1215 015346                                    PRINTF  @TFM36,ADDR,UNIT
1216 015376  004737  007604                    CALL    CHKMAX
1217 015402                            ENDMSG
1218
1219 015404     045     116     045    TFM36:  .ASCIZ  /%N%A REGISTER ADDRESS ERROR, ADDRESS = %06%A, UNIT = %02%N/
1220                                    .EVEN
1221
```

```
1223
1224                                    ;++
1225                                    ; ERROR REPORT FOR  KMV11 CSR TEST
1226                                    ;--
1227
1228 015500                             BGNMSG  PRSEL
1229 015500                                     PRINTB  @MSFL,NUM0,BAD,GOOD
1230 015534  004737  007604                      CALL    CHKMAX
1231 015540                             ENDMSG
1232
1233 015542                             BGNMSG  PRBAD
1234 015542                                     PRINTB  @MBAD,NUM0,BAD
1235 015572  004737  007604                      CALL    CHKMAX
1236 015576                             ENDMSG
1237
1238 015600                             BGNMSG  PRREG
1239                                    ;+
1240                                    ; INPUT: NUM0 CONTAINS THE OFFSET OF THE BAD LOCATION
1241                                    ;        NUM1 CONTAINS THE OFFSET OF THE SELECTED LOCATION
1242                                    ;        DATA CONTAINS THE CURRENT PATTERN
1243                                    ;        BAD  CONTAINS THE BAD VALUE
1244                                    ;        GOOD CONTAINS THE GOOD VALUE
1245                                    ;        R1   POINTS ON THE CURRENT OFFSET IN THE SOURCE BUFFER
1246                                    ;        R2   POINTS ON THE CURRENT OFFSET IN THE DESTINATION BUFFER
1247                                    ;-
1248 015600                                     PRINTB  @MPAT,NUM1,DATA
1249 015630                                     PRINTB  @MSEL,NUM0,BAD,GOOD
1250 015664  062737  000002  002402  2$:        ADD     @2,NUM0
1251 015672  023727  002402  000040             CMP     NUM0,@40
1252 015700  001423                             BEQ     1$
1253 015702  012137  002350                      MOV     (R1)+,GOOD
1254 015706  012237  002352                      MOV     (R2)+,BAD
1255 015712                                     PRINTX  @MSEL,NUM0,BAD,GOOD
1256 015746  000746                             BR      2$
1257 015750  000240                     1$:     NOP
1258 015752  004737  007604                      CALL    CHKMAX
1259 015756                             ENDMSG
1260
1261 015760                             BGNMSG  PRALL
1262                                    ;+
1263                                    ; INPUT: NUM0 CONTAINS THE OFFSET OF THE BAD LOCATION
1264                                    ;        BAD  CONTAINS THE BAD VALUE
1265                                    ;        GOOD CONTAINS THE GOOD VALUE
1266                                    ;        R2   POINTS ON THE CURRENT LOCATION IN THE BUFFER
1267                                    ;-
1268 015760                                     PRINTB  @MSEL,NUM0,BAD,GOOD
1269 016014  062737  000002  002402  2$:        ADD     @2,NUM0
1270 016022  022737  000040  002402             CMP     @40,NUM0
1271 016030  001421                             BEQ     1$
1272 016032  012237  002352                      MOV     (R2)+,BAD
1273 016036                                     PRINTX  @MSEL,NUM0,BAD,GOOD
1274 016072  000750                             BR      2$
1275 016074  000240                     1$:     NOP
1276 016076  004737  007604                      CALL    CHKMAX
1277 016102                             ENDMSG
1278
1279 016104     045     116     045     MSEL:   .ASCIZ  /%N%A SEL%02%A = %06%A   EXPECTED VALUE = %06/
```

```
1280 016161     045     116     045  MBAD:   .ASCIZ  /%N%A SEL%O2%A READ = %O6/
1281 016212     045     116     045  MPAT:   .ASCIZ  /%N%A SEL%O2%A UNDER TEST FOR PATTERN = %O6/
1282                                         .EVEN
```

```
                                        ;/V01.01/
                                        ;
                                        ; ERROR ROUTINE FOR STACK PROBLEM IN MAINT2
                                        ; INPUT: BAD CONTAINS THE BAD VALUE
                                        ;        GOOD CONTAINS THE EXPECTED ONE
                                        ;        SPADDR THE STACK ADDRESS
                                        ;
016266                                  BGNMSG  PRSTAK
016266                                          PRINTB  @MSTACK,SPADDR,BAD,GOOD
016322  004737  007604                          CALL    CHKMAX
016326                                  ENDMSG
                                        ;
                                        ; ERROR ROUTINE FOR CHECKSUM ERROR
                                        ; INPUT: BAD CONTAINS THE WRONG CHECKSUM RETURNED BY THE FW
                                        ;
016330                                  BGNMSG  PRCHEK
016330                                          PRINTB  @MSCHEK,BAD
016354  004737  007604                          CALL    CHKMAX
016360                                  ENDMSG

016362      045      116      045  MSTACK: .ASCIZ  /%N%A LOCATION = %06%A CONTAINS VALUE = %06%A WHEN EXPECTED VALUE= %06/
016470      045      116      045  MSCHEK: .ASCIZ  /%N%A OBTAINED CHECKSUM = %06/
                                            .EVEN
```

```
                                        ;++
                                        ; ERROR REPORT FOR REVISION CODE
                                        ;--

016526                                  BGNMSG  PRREV
016526  010137  002352                        MOV     R1,BAD
016532                                        PRINTB  @MREV,BAD,REVCOD
016562  004737  007604                        CALL    CHKMAX
016566                                  ENDMSG

016570     045     116     045  MREV:   .ASCIZ  /%N%A REVISION CODE = %06%A  EXPECTED ONE = %06/
                                        .EVEN
```

```
1325
1326
1327
1328                                    ;**
1329                                    ; ERROR REPORT FOR KMV11 RAM TEST
1330                                    ;--
1331
1332 016650                            BGNMSG  PRRAM0
1333 016650                                    PRINTB  @MRAM0,TSTLOC,DATA
1334 016700                                    PRINTB  @MRAM1,BAD,BADLOC
1335 016730  004737  007604                     CALL    CHKMAX
1336 016734                            ENDMSG
1337
1338 016736                            BGNMSG  PRRAM1
1339 016736                                    PRINTB  @MRAM1,BAD,BADLOC
1340 016766  004737  007604                     CALL    CHKMAX
1341 016772                            ENDMSG
1342
1343 016774                            BGNMSG  PRRAM2
1344 016774                                    PRINTB  @MRAM2,TSTLOC,BDDAT,DATA
1345 017030  004737  007604                     CALL    CHKMAX
1346 017034                            ENDMSG
1347
1348 017036     045     116     045   MRAM0:  .ASCIZ  /%N%A LOCATION UNDER TEST = %O6%A  FOR PATTERN = %O6/
1349                                    .EVEN
1350 017122     045     116     045   MRAM1:  .ASCIZ  /%N%A BAD VALUE = %O6%A  AT LOCATION = %O6/
1351                                    .EVEN
1352 017174     045     116     045   MRAM2:  .ASCII  /%N%A LOCATION UNDER TEST = %O6/
1353 017232     045     116     045           .ASCIZ  /%N%A RAM VALUE = %O6%A   EXPECTED VALUE = %O6/
1354                                    .EVEN
```

```
1356
1357
1358
1359                                    ;**
1360                                    ; ERROR REPORT ON Q-BUS INTERRUPT TEST
1361                                    ;--
1362
1363 017310                            BGNMSG  PRLVL
1364 017310                                    PRINTB  @MLVL,GDVEC,BDLVL,GDLVL
1365 017344  004737  007604                     CALL    CHKMAX
1366 017350                            ENDMSG
1367
1368
1369 017352     045     116     045    MLVL:   .ASCII  /%N%A INTERRUPT VECTOR = %03/
1370 017405     045     116     045            .ASCIZ  /%N%A AT PRIORITY LEVEL = %03%A  EXPECTED  = %03/
1371                                    .EVEN
1372
1373 017466                            BGNMSG  PRVEC
1374 017466                                    PRINTB  @MVEC,GDVEC
1375 017512  004737  007604                     CALL    CHKMAX
1376 017516                            ENDMSG
1377
1378 017520     045     116     045    MVEC:   .ASCIZ  /%N%A EXPECTED INTERRUPT VECTOR = %03/
1379                                    .EVEN
```

```
1381
1382
1383
1384
1385                                    ;++
1386                                    ; ERROR REPORT FOR INTERRUPT ON CSR WRITE ACCESS
1387                                    ;--
1388
1389 017566                            BGNMSG  PRCSR
1390 017566                                    PRINTB  #MCSR,NUM1
1391 017612  004737  007604                    CALL    CHKMAX
1392 017616                            ENDMSG
1393
1394 017620     045     116     045    MCSR:   .ASCIZ  /%N%A   CSR ACCESSED - SEL%02/
1395                                    .EVEN
1396
1397
1398 017654                            BGNMSG  PGCSR
1399 017654                                    PRINTB  #MGCSR,NUM0
1400 017700  004737  007604                    CALL    CHKMAX
1401 017704                            ENDMSG
1402
1403 017706     045     116     045    MGCSR:  .ASCIZ  /%N%A SEL%01%A ACCESS UNDER TEST/
1404                                    .EVEN
```

```
1406
1407
1408                                    ;++
1409                                    ; ERROR REPORT FOR RT CLOCK
1410                                    ;--
1411
1412 017746                            BGNMSG  PRPER
1413 017746                                    PRINTB  @MPER,NUM0
1414 017772  004737  007604                    CALL    CHKMAX
1415 017776                            ENDMSG
1416
1417 020000     045     116     045    MPER:   .ASCIZ  /%N%A   DURING CLOCK PERIOD = %01/
1418                                    .EVEN
```

```
1420
1421
1422                                    ;++
1423                                    ; ERROR REPORT FOR LINE CLOCK
1424                                    ;--
1425
1426 020040                            BGNMSG  PRRATE
1427 020040  004737  020052                    CALL    PRIRAT
1428 020044  004737  007604                    CALL    CHKMAX
1429 020050                            ENDMSG
1430
1431 020052  016137  020106  002352    PRIRAT: MOV     RATETB(R1),BAD
1432 020060                                    PRINTB  #MRATE,BAD
1433 020104  000207                            RETURN
1434
1435 020106  020124                    RATETB: RATE0
1436 020110  020132                            RATE1
1437 020112  020140                            RATE2
1438 020114  020147                            RATE3
1439 020116  020154                            RATE4
1440 020120  020161                            RATE5
1441 020122  020166                            RATE6
1442
1443 020124     040     062     056    RATE0:  .ASCIZ  / 2.4K/
1444 020132     040     071     056    RATE1:  .ASCIZ  / 9.6K/
1445 020140     040     061     071    RATE2:  .ASCIZ  / 19.2K/
1446 020147     040     064     070    RATE3:  .ASCIZ  / 48K/
1447 020154     040     065     066    RATE4:  .ASCIZ  / 56K/
1448 020161     040     066     064    RATE5:  .ASCIZ  / 64K/
1449 020166     040     067     062    RATE6:  .ASCIZ  / 72K/
1450
1451 020173     045     116     045    MRATE:  .ASCIZ  /%N%A SELECTED RATE = %T/
1452                                    .EVEN
```

```
                                        ;++
                                        ; ERROR REPORT FOR TEST17
                                        ;--
020224                                  BGNMSG  PRITRX
020224  004737  020052                          CALL    PRIRAT                  ;PRINT CLOCK RATE
020230  016537  000004  002360                  MOV     SEL4(R5),DATA           ;GET INTERRUPT COUNTER
020236                                          PRINTB  #MTMOTX,DATA            ;PRINT INTERRUPT COUNT NUMBER
020262  004737  007604                          CALL    CHKMAX
020266                                  ENDMSG

020270     045     116     045  MTMOTX: .ASCIZ  /%N%A INTERRUPT COUNT = %D5/
                                        .EVEN
```

```
                                ;++
                                ; ERROR REPORT FOR RCV TESTS
                                ;
                                ; INPUT: R1 CONTAINS THE WORD OFFSET OF LINE RATE VALUE IN A TABLE
                                ;        BDDAT CONTAINS THE RCV STATUS AND DATA REGISTER VALUE
                                ;        GDDAT CONTAINS THE CORRESPONDING EXPECTED VALUE OF BDDAT
                                ;        SEL10 CONTAINS THE FLAG WORD
                                ;--

020324                          BGNMSG  PRDAT
020324  004737  020362                  CALL    PRDATG                  ;CALL COMMON SUBROUTINE
020330                                  PRINTX  #MEVT
020350  004737  020446                  CALL    COMFLG                  ;PRINT FLAGWORD IN DETAIL
020354  004737  007604                  CALL    CHKMAX
020360                          ENDMSG

020362  016137  020106  002352  PRDATG: MOV     RATETB(R1),BAD
020370                                  PRINTB  #MRATE,BAD              ;PRINT CLOCK RATE
020414                                  PRINTB  #MDAT,BDDAT,GDDAT       ;PRINT STATUS&DATA REGISTER
020444  000207                          RETURN

020446                          COMFLG: ;DISPLAY BIT PER BIT THE FLAGWORD CONTAINED IN SEL10

020446  012701  002430                  MOV     #BUF01,R1
020452  016502  000010                  MOV     SEL10(R5),R2
020456  012703  000001                  MOV     #BIT0,R3
020462  005011                  1$:     CLR     (R1)
020464  030302                          BIT     R3,R2
020466  001401                          BEQ     2$
020470  005211                          INC     (R1)
020472                          2$:
020472  005721                          TST     (R1)+
020474  006303                          ASL     R3
020476  001371                          BNE     1$
020500                                  PRINTX  #EVT0,BUF01             ;BIT0
020524                                  PRINTX  #EVT1,BUF01+2           ;BIT1
020550                                  PRINTX  #EVT2,BUF01+4           ;BIT2
020574                                  PRINTX  #EVT3,BUF01+6           ;BIT3
020620                                  PRINTX  #EVT4,BUF01+10          ;BIT4
020644                                  PRINTX  #EVT7,BUF01+16          ;BIT7
020670                                  PRINTX  #EVT8,BUF01+20          ;BIT8
020714                                  PRINTX  #EVT9,BUF01+22          ;BIT9
020740                                  PRINTX  #EVT10,BUF01+24         ;BIT10
020764                                  PRINTX  #EVT11,BUF01+26         ;BIT11
021010                                  PRINTX  #EVT12,BUF01+30         ;BIT12
021034                                  PRINTX  #EVT13,BUF01+32         ;BIT13
021060                                  PRINTX  #EVT15,BUF01+36         ;BIT15
021104                                  PRINTB  #MNULL
021124  000207                          RETURN

021126    045   116   045   MDAT:   .ASCIZ  /%N%A OBTAINED STATUS&DATA = %O6%A  EXPECTED ONE = %O6/
021214    045   116   045   MNULL:  .ASCIZ  /%N%A    %N/
021227    045   116   045   MEVT:   .ASCIZ  /%N%A CURRENT EVENT FLAG SETTING:/
021270    045   116   045   EVT0:   .ASCIZ  /%N%A- XMT DATA LOADING STARTED        BIT = %B1/
021350    045   116   045   EVT1:   .ASCIZ  /%N%A- ALL DATA TRANSMITTED           BIT = %B1/
```

```
1526 021430     045     116     045  EVT2:   .ASCIZ  /%N%A- TRX DATA INT. WHILE DISABLED    BIT = %B1/
1527 021510     045     116     045  EVT3:   .ASCIZ  /%N%A- UNDERRUN INTERRUPT               BIT = %B1/
1528 021570     045     116     045  EVT4:   .ASCIZ  /%N%A- UNDERRUN INTERRUPT AFTER ACK     BIT = %B1/
1529 021650     045     116     045  EVT7:   .ASCIZ  /%N%A- UNEXPECTED STATUS RECEIVED       BIT = %B1/
1530 021730     045     116     045  EVT8:   .ASCIZ  /%N%A- FIRST EXPECTED STATUS RECEIVED  BIT = %B1/
1531 022010     045     116     045  EVT9:   .ASCIZ  /%N%A- LAST EXPECTED STATUS RECEIVED   BIT = %B1/
1532 022070     045     116     045  EVT10:  .ASCIZ  /%N%A- UNEXPECTED RCV DATA              BIT = %B1/
1533 022150     045     116     045  EVT11:  .ASCIZ  /%N%A- FIRST DATA RECEIVED              BIT = %B1/
1534 022230     045     116     045  EVT12:  .ASCIZ  /%N%A- LAST DATA RECEIVED               BIT = %B1/
1535 022310     045     116     045  EVT13:  .ASCIZ  /%N%A- RCV DATA INT. WHILE DISABLED    BIT = %B1/
1536 022370     045     116     045  EVT15:  .ASCIZ  /%N%A- INTERRUPT AT ILLEGAL VECTOR     BIT = %B1/
1537                                  .EVEN
```

```
1539                                    ;++
1540                                    ; PREVT:
1541                                    ;
1542                                    ; INPUT: R1 CONTAINS THE WORD OFFSET OF LINE RATE VALUE IN A TABLE
1543                                    ;        SEL10 CONTAINS THE FLAG WORD
1544                                    ;--
1545 022450                            BGNMSG  PREVT
1546 022450  004737  022462                    CALL    PREVTG
1547 022454  004737  007604                    CALL    CHKMAX
1548 022460                            ENDMSG
1549
1550 022462  016137  020106  002352  PREVTG: MOV     RATETB(R1),BAD
1551 022470                                    PRINTB  #MRATE,BAD
1552 022514                                    PRINTB  #MEVT
1553 022534  004737  020446                    CALL    COMFLG
1554 022540  000207                            RETURN
1555                                    .EVEN
```

```
1557                                    ;**
1558                                    ; ERROR REPORT DURING TRANSMIT DMA
1559                                    ;
1560                                    ; PDMATG:
1561                                    ; INPUT: SEL12 :CURRENT TRX BYTE COUNT
1562                                    ;        SEL14 AND 16 : CURRENT TRX BUFFER ADDRESS
1563                                    ;
1564                                    ;--
1565 022542                            PDMATG: PRINTB  @DMATG1,SEL12(R5)
1566 022566  042765  177700  000016            BIC     #177700,SEL16(R5)               ;KEEP ONLY VALID BITS OF
1567                                                                                    ;EXTENDED ADDRESS
1568 022574                                    PRINTB  @DMATG2,SEL14(R5),SEL16(R5)
1569 022624                                    PRINTB  @MNULL
1570 022644  000207                            RETURN
1571
1572 022646                            BGNMSG  PDMAT1
1573 022646  004737  022542                    CALL    PDMATG
1574 022652  004737  007604                    CALL    CHKMAX
1575 022656                            ENDMSG
1576
1577 022660                            BGNMSG  PDMAT2
1578 022660  004737  020362                    CALL    PRDATG          ;PRINT STATUS&DATA REGISTER
1579 022664  004737  022542                    CALL    PDMATG          ;PRINT TRX DMA REGISTERS
1580 022670  004737  007604                    CALL    CHKMAX
1581 022674                            ENDMSG
1582
1583 022676                            BGNMSG  PDMAT3
1584 022676  004737  022714                    CALL    PEVTTG          ;PRINT FLAGWORD
1585 022702  004737  022542                    CALL    PDMATG          ;PRINT TRX DMA REGISTERS
1586 022706  004737  007604                    CALL    CHKMAX
1587 022712                            ENDMSG
1588
1589 022714  016137  020106  002352    PEVTTG: MOV     RATETB(R1),BAD
1590 022722                                    PRINTB  @MRATE,BAD
1591 022746  012701  002430                    MOV     #BUF01,R1
1592 022752  016502  000010                    MOV     SEL10(R5),R2
1593 022756  012703  000001                    MOV     #BIT0,R3
1594 022762  005011                    1$:     CLR     (R1)
1595 022764  030302                            BIT     R3,R2
1596 022766  001401                            BEQ     2$
1597 022770  005211                            INC     (R1)
1598 022772                            2$:
1599 022772  005721                            TST     (R1)+
1600 022774  006303                            ASL     R3
1601 022776  001371                            BNE     1$
1602 023000                                    PRINTB  @MEVT
1603 023020                                    PRINTX  @EVT0,BUF01             ;BIT0
1604 023044                                    PRINTX  @EVT1,BUF01+2           ;BIT1
1605 023070                                    PRINTX  @EVTT2,BUF01+4          ;BIT2
1606 023114                                    PRINTX  @EVB3,BUF01+6           ;BIT3
1607 023140                                    PRINTX  @EVB10,BUF01+10         ;BIT4
1608 023164                                    PRINTX  @EVB2,BUF01+12          ;BIT5
1609 023210                                    PRINTX  @EVT7,BUF01+16          ;BIT7
1610 023234                                    PRINTX  @EVT8,BUF01+20          ;BIT8
1611 023260                                    PRINTX  @EVT9,BUF01+22          ;BIT9
1612 023304                                    PRINTX  @EVR10,BUF01+24         ;BIT10
1613 023330                                    PRINTX  @EVT11,BUF01+26         ;BIT11
```

```
1614 023354                                    PRINTX  #EVT12,BUF01+30          ;BIT12
1615 023400                                    PRINTX  #EVT13,BUF01+32          ;BIT13
1616 023424                                    PRINTX  #EVT15,BUF01+36          ;BIT15
1617 023450                                    PRINTB  #MNULL
1618 023470  000207                            RETURN
1619
1620 023472     045     116     045  EVTT2:    .ASCIZ  /%N%A- UNEXPECTED END OF RCV DMA       BIT = %B1/
1621
1622 023552     045     116     045  DMATG1:   .ASCIZ  /%N%A CURRENT TRANSMIT DMA BYTE COUNT = %O6/
1623 023625     045     116     045  DMATG2:   .ASCII  /%N%A CURRENT TRANSMIT DMA BUFFER ADDRESS = %O6/
1624 023703     045     101     040            .ASCIZ  /%A AND EXTENDED BITS = %O6/
1625                                           .EVEN
```

```
                                        ;**
                                        ; ERROR REPORT DURING RECEIVE DMA
                                        ;
                                        ; INPUT: SEL12 :CURRENT RCV BYTE COUNT
                                        ;        SEL14 AND 16 : CURRENT RCV BUFFER ADDRESS
                                        ;
                                        ;--
023736                                  PDMARG:
023736                                          PRINTB  #DMARG1,SEL12(R5)
023762  042765  177700  000016                  BIC     #177700,SEL16(R5)               ;KEEP ONLY VALID BITS OF
                                                                                        ;EXTENDED ADDRESS
023770                                          PRINTB  #DMARG2,SEL14(R5),SEL16(R5)
024020                                          PRINTB  #MNULL
024040  000207                                  RETURN

024042                                  BGNMSG  PDMAR1
024042  004737  023736                          CALL    PDMARG          ;PRINT RCV DMA REGISTERS
024046  004737  007604                          CALL    CHKMAX
024052                                  ENDMSG

024054                                  BGNMSG  PDMAR2
024054  004737  020362                          CALL    PRDATG          ;PRINT STATUS&DATA REGISTER
024060  004737  023736                          CALL    PDMARG          ;PRINT RCV DMA REGISTERS
024064  004737  007604                          CALL    CHKMAX
024070                                  ENDMSG

024072                                  BGNMSG  PDMAR3
024072  004737  024110                          CALL    PEVTRG          ;PRINT FLAGWORD
024076  004737  023736                          CALL    PDMARG          ;PRINT RCV DMA REGISTERS
024102  004737  007604                          CALL    CHKMAX
024106                                  ENDMSG

024110  016137  020106  002352        PEVTRG: MOV     RATETB(R1),BAD
024116                                          PRINTB  #MRATE,BAD
024142  012701  002430                          MOV     #BUF01,R1
024146  016502  000010                          MOV     SEL10(R5),R2
024152  012703  000001                          MOV     #BIT0,R3
024156  005011                          1$:     CLR     (R1)
024160  030302                                  BIT     R3,R2
024162  001401                                  BEQ     2$
024164  005211                                  INC     (R1)
024166                                  2$:
024166  005721                                  TST     (R1)+
024170  006303                                  ASL     R3
024172  001371                                  BNE     1$
024174                                          PRINTB  #MEVT
024214                                          PRINTX  #EVT0,BUF01             ;BIT0
024240                                          PRINTX  #EVT1,BUF01+2           ;BIT1
024264                                          PRINTX  #EVR2,BUF01+4           ;BIT2
024310                                          PRINTX  #EVB4,BUF01+6           ;BIT3
024334                                          PRINTX  #EVB10,BUF01+10         ;BIT4
024360                                          PRINTX  #EVB5,BUF01+12          ;BIT5
024404                                          PRINTX  #EVT7,BUF01+16          ;BIT7
024430                                          PRINTX  #EVT8,BUF01+20          ;BIT8
024454                                          PRINTX  #EVT9,BUF01+22          ;BIT9
024500                                          PRINTX  #EVT15,BUF01+36         ;BIT15
024524                                          PRINTB  #MNULL
```

```
1684 024544  000207                          RETURN
1685
1686 024546     045     116     045  EVR2:   .ASCIZ  /%N%A- UNEXPECTED END OF TRX DMA       BIT = %B1/
1687 024626     045     116     045  EVR10:  .ASCIZ  /%N%A- DATA COMPARE ERROR               BIT = %B1/
1688
1689 024706     045     116     045  DMARG1: .ASCIZ  /%N%A CURRENT RECEIVE  DMA BYTE COUNT = %O6/
1690 024761     045     116     045  DMARG2: .ASCII  /%N%A CURRENT RECEIVE DMA BUFFER ADDRESS = %O6/
1691 025036     045     101     040          .ASCIZ  /%A AND EXTENDED BITS = %O6/
1692                                         .EVEN
```

```
                                        ;++
                                        ; ERROR REPORT DATA COMPARE ERROR DURING RECEIVE DMA
                                        ;
                                        ; INPUT: BAD CONTAINS THE BAD VALUE RECEIVED IN THE RCVBUF
                                        ;        GOOD CONTAINS THE EXPECTED VALUE
                                        ;        R1 POINTS ON THE PATTERN BUFFER, WHICH IS THE REFERENCE BUFFER
                                        ;        R2 POINTS ON NEXT LOCATION IN THE RECEIVE BUFFER
                                        ;        R3 CONTAINS THE END REFERENCE BUFFER ADDRESS
                                        ;--
025072                                  BGNMSG  PRDMAR
025072                                          PRINTB  @DMAR1,BAD,GOOD
025122  112137  002350                  1$:     MOVB    (R1)+,GOOD              ;NEXT EXPECTED VALUE
025126  112237  002352                          MOVB    (R2)+,BAD               ;NEXT RECEIVED VALUE
025132  020301                                  CMP     R3,R1                   ;END OF BUFFER ?
025134  002415                                  BLT     2$                      ;YES
025136                                          PRINTB  @DMAR2,BAD,GOOD
025166  000755                                  BR      1$
025170  000240                  2$:     NOP
025172                                          PRINTB  @MNULL
025212  004737  007604                          CALL    CHKMAX
025216                                  ENDMSG
025220      045     116     045         DMAR1:  .ASCIZ  /%N%A BAD VALUE =     %03%A      EXPECTED VALUE = %03/
025305      045     116     045         DMAR2:  .ASCIZ  /%N%A NEXT LOCATION = %03%A      EXPECTED VALUE = %03/
                                                .EVEN
```

```
1720                                    ;**
1721                                    ; ERROR REPORT DURING DMA TRANSFER IN BOTH DIRECTIONS
1722                                    ;
1723                                    ; INPUT: SEL12 :CURRENT TRX BYTE COUNT
1724                                    ;        SEL14 AND 16 : CURRENT TRX BUFFER ADDRESS
1725                                    ;        SEL20 :CURRENT RCV BYTE COUNT
1726                                    ;        SEL22 AND 24: CURRENT RCV BUFFER ADDRESS
1727                                    ;--
1728 025372                            PDMRTG: PRINTB  #DMATG1,SEL12(R5)
1729 025416  042765  177700  000016            BIC     #177700,SEL16(R5)               ;KEEP ONLY VALID BITS IN
1730                                                                                    ;EXTENDED ADDRESS
1731 025424                                    PRINTB  #DMATG2,SEL14(R5),SEL16(R5)
1732 025454                                    PRINTB  #DMARG1,SEL20(R5)
1733 025500  042765  177700  000024            BIC     #177700,SEL24(R5)               ;KEEP ONLY VALID BITS IN
1734                                                                                    ;EXTENDED ADDRESS
1735 025506                                    PRINTB  #DMARG2,SEL22(R5),SEL24(R5)
1736 025536                                    PRINTB  #MNULL
1737 025556  000207                            RETURN
1738
1739 025560                            BGNMSG  PDMART
1740 025560  004737  025372                    CALL    PDMRTG
1741 025564  004737  007604                    CALL    CHKMAX
1742 025570                            ENDMSG
1743
1744 025572                            BGNMSG  PDMAB1
1745 025572  004737  020362                    CALL    PRDATG          ;PRINT STATUS&DATA REGISTER
1746 025576  004737  025372                    CALL    PDMRTG          ;PRINT TRX AND RCV DMA REGISTERS
1747 025602  004737  007604                    CALL    CHKMAX
1748 025606                            ENDMSG
1749
1750 025610                            BGNMSG  PDMAB2
1751 025610  004737  025626                    CALL    PEVTBG          ;PRINT FLAGWORD
1752 025614  004737  025372                    CALL    PDMRTG          ;PRINT TRX AND RCV DMA REGISTERS
1753 025620  004737  007604                    CALL    CHKMAX
1754 025624                            ENDMSG
1755
1756 025626  016137  020106  002352    PEVTBG: MOV     RATETB(R1),BAD
1757 025634                                    PRINTB  #MRATE,BAD
1758 025660  012701  002430                    MOV     #BUF01,R1
1759 025664  016502  000010                    MOV     SEL10(R5),R2
1760 025670  012703  000001                    MOV     #BIT0,R3
1761 025674  005011                    1$:     CLR     (R1)
1762 025676  030302                            BIT     R3,R2
1763 025700  001401                            BEQ     2$
1764 025702  005211                            INC     (R1)
1765 025704                            2$:
1766 025704  005721                            TST     (R1)+
1767 025706  006303                            ASL     R3
1768 025710  001371                            BNE     1$
1769 025712                                    PRINTB  #MEVT
1770 025732                                    PRINTX  #EVT0,BUF01             ;BIT0
1771 025756                                    PRINTX  #EVT1,BUF01+2           ;BIT1
1772 026002                                    PRINTX  #EVB2,BUF01+4           ;BIT2
1773 026026                                    PRINTX  #EVB3,BUF01+6           ;BIT3
1774 026052                                    PRINTX  #EVB4,BUF01+10          ;BIT4
1775 026076                                    PRINTX  #EVB5,BUF01+12          ;BIT5
1776 026122                                    PRINTX  #EVT7,BUF01+16          ;BIT7
```

```
1777 026146                                    PRINTX  @EVT8,BUF01+20          ;BIT8
1778 026172                                    PRINTX  @EVT9,BUF01+22          ;BIT9
1779 026216                                    PRINTX  @EVB10,BUF01+24         ;BIT10
1780 026242                                    PRINTX  @EVT15,BUF01+36         ;BIT15
1781 026266                                    PRINTB  @MNULL
1782 026306  000207                            RETURN
1783 026310     045     116     045  EVB2:     .ASCIZ  /%N%A- TIMEOUT DURING TRANSMIT DMA     BIT = %B1/
1784 026370     045     116     045  EVB3:     .ASCIZ  /%N%A- END OF TRX DMA WITHOUT BCNTIN   BIT = %B1/
1785 026450     045     116     045  EVB4:     .ASCIZ  /%N%A- OVERFLOW DURING RCV DMA         BIT = %B1/
1786 026530     045     116     045  EVB5:     .ASCIZ  /%N%A- TIMEOUT DURING RECEIVE DMA      BIT = %B1/
1787 026610     045     116     045  EVB10:    .ASCIZ  /%N%A- UNDERRUN INTERRUPT OCCURS       BIT = %B1/
1788
```

```
1790                                    ;**
1791                                    ; ERROR REPORT FOR MODEM SIGNAL TEST
1792                                    ;--
1793
1794 026670                            BGNMSG  PRMDM
1795 026670                                    PRINTB  @MMODO,DATA
1796 026714                                    PRINTB  @MMOD1,BAD,GOOD
1797 026744  004737  007604                    CALL    CHKMAX
1798 026750                            ENDMSG
1799
1800 026752     045     116     045   MMODO:  .ASCIZ  /%N%A   MODEM CONTROL BITS SET IN PORT B = %B8/
1801 027026     045     116     045   MMOD1:  .ASCII  /%N%A   MODEM CONTROL BITS READ IN PORT A = %B8/
1802 027102     045     101     040           .ASCIZ  /%A  EXPECTED BITS  = %B8/
1803
1804                                    .EVEN
```

```
1806
1807
1808
1809
1810                                    ;++
1811                                    ; ERROR REPORT FOR KMV11 R/W INTERNAL REGISTERS
1812                                    ;--
1813
1814 027134                            BGNMSG  INREG
1815 027134  013702  002352                    MOV     BAD,R2
1816 027140  005742                            TST     -(R2)
1817 027142  016237  027230  027226            MOV     REGTB(R2),REGIN
1818 027150                                    PRINTB  REGIN
1819 027170                                    PRINTB  #INDAT,BDDAT,DATA
1820 027220  004737  007604                    CALL    CHKMAX
1821 027224                            ENDMSG
1822
1823 027226  000000                    REGIN:  0
1824
1825 027230  027244  027276  027331   REGTB:  INRG1,INRG2,INRG3,INRG4,INRG5,INRG6
1826
1827 027244      045     116     045   INRG1:  .ASCIZ  /%N%A BYTE COUNT OUT ERROR/
1828 027276      045     116     045   INRG2:  .ASCIZ  /%N%A BUS ADDRESS OUT ERROR/
1829 027331      045     116     045   INRG3:  .ASCIZ  /%N%A EXT. ADDRESS OUT ERROR/
1830 027365      045     116     045   INRG4:  .ASCIZ  /%N%A BYTE COUNT IN ERROR/
1831 027416      045     116     045   INRG5:  .ASCIZ  /%N%A BUS ADDRESS IN ERROR/
1832 027450      045     116     045   INRG6:  .ASCIZ  /%N%A EXT. ADDRESS IN ERROR/
1833
1834 027503      045     101     040   INDAT:  .ASCIZ  /%A DATA READ = %06%A   EXPECTED DATA = %06/
1835                                    .EVEN
1836
1837
1838
```

```
                                                .SBTTL  INITIALIZE SECTION

                                                ;++
                                                ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                                ; AT THE BEGINNING OF EACH PASS.
                                                ;--

 027556                                                 BGNINIT


                                                .IF NE HELP
                                                ;********************************************************************************
                                                ;       THE INITIALIZE CODE IS EXECUTED UNDER FIVE CONDITIONS.  THERE
                                                ;       ARE SUPERVISOR EVENT FLAGS THAT ARE USED TO LET THE
                                                ;       DIAGNOSTIC KNOW UNDER WHICH CONDITION THE EXECUTION IS TAKING
                                                ;       PLACE.  THE EVENT FLAGS ARE READ USING THE "READEF" MACRO.
                                                ;       THE CONDITIONS UNDER WHICH THE INIT CODE IS EXECUTED AND THE
                                                ;       CORRRESPONDING EVENT FLAGS ARE:
                                                ;               START COMMAND           EF.START
                                                ;               RESTART COMMAND         EF.RESTART
                                                ;               CONTINUE COMMAND        EF.CONTINUE
                                                ;               POWERDOWN/POWERUP       EF.PWR
                                                ;               NEW PASS                EF.NEW
                                                ;       EXAMPLE OF EVENT FLAG USE:
                                                ;               READEF  #EF.START
                                                ;               BCOMPLETE       STARTCODE
                                                ;       DURING THE INIT CODE, USE THE "GPHARD" MACRO TO OBTAIN P-TABLE
                                                ;       INFORMATION FOR DEVICE TESTING.  GET ONE UNIT'S INFORMATION IF
                                                ;       THIS IS A SEQUENTIAL DIAGNOSTIC.  GET INFORMATION ON ALL
                                                ;       UNITS AVAILABLE FOR TESTING IF THIS IS AN EXERCISER.  THE NUMBER
                                                ;       OF UNITS AVAILABLE IS IN A HEADER LOCATION: "L$UNIT".
                                                ;********************************************************************************
                                                .ENDC
                                                .IF NE HELP
                                                ;********************************************************************************
                                                ;     INSERT LOCAL STORAGE THAT IS USED ONLY
                                                ;     DURING THE INITIALIZE SECTION.
                                                ;********************************************************************************

                                                ;********************************************************************************
                                                ;     INSERT MESSAGES THAT ARE USED ONLY
                                                ;     DURING THE INITIALIZE SECTION.
                                                ;********************************************************************************
                                                .ENDC

                                                        .EVEN





 027556  005737  002340                                 TST     FTIME                   ;IS THE FIRST PASS?
 027562  001011                                         BNE     1$                      ;IF NOT
 027564  013737  000004  002344                         MOV     @#4,SAVE4               ;SAVE TIME-OUT TRAP VECTOR IF YES
 027572  013737  000006  002346                         MOV     @#6,SAVE6
 027600  012737  000001  002340                         MOV     #1,FTIME                ;THEN SET FIRST PASS FLAG

 027606  013737  002344  000004         1$:             MOV     SAVE4,@#4               ;RESTORE TRAP VECTOR
```

INITIALIZE SECTION

```
1897 027614  013737  002346  000006          MOV     SAVE6,@#6
1898
1899                                 ;SEE IF PROGRAM WAS JUST CONTINUE       ; /V01.04/
1900 027622                          READEF  #EF.CONTINUE
1901 027630                          BCOMPLETE       ENDINI
1902
1903                                 ;SEE IF PROGRAM JUST STARTED, BR IF YES
1904 027632                          READEF  #EF.START
1905 027640                          BCOMPLETE       SETUP
1906
1907                                 ;SEE IF THIS IS A NEW PASS, BR IF YES
1908 027642                          READEF  #EF.NEW
1909 027650                          BNCOMPLETE      NEXT
1910
1911
1912 027652                          SETUP:
1913 027652  012737  177777  002514          MOV     #-1,UUT         ;INIT UNIT NUMBER
1914 027660                          NEXT:
1915                                 ;TEST IF ALL UNITS TESTED
1916 027660  005237  002514          INC     UUT
1917
1918                                 ;GET P-TABLE IF AVAILABLE FOR THIS UNIT
1919 027664                          GPHARD  UUT,R1
1920 027674                          BNCOMPLETE      NEXT            ;IF NOT, TRY THE NEXT ONE
1921
1922                                 ;GET KMV11-C CSR ADDRESS
1923 027676  012137  002474          MOV     (R1)+,KMVCSR            ;LOAD CSR ADDRESS
1924
1925                                 ;GET KMV11 INTERRUPT VECTORS
1926 027702  011137  002476          MOV     (R1),KMVV00
1927 027706  012137  002500          MOV     (R1)+,KMVV04
1928 027712  062737  000004  002500  ADD     #4,KMVV04
1929
1930                                 ;GET KMV11 PRIORITY
1931 027720  012137  002502          MOV     (R1)+,KMVLVL
1932
1933                                 ;GET LOOP INDICATOR
1934 027724  012137  002366          MOV     (R1)+,LCLOOP
1935
1936                                 ;GET KMV11 LINE CLOCK RATE
1937 027730  012137  002362          MOV     (R1)+,RATE
1938
1939 027734                          10$:
1940
1941                                 ; LONG RAM TEST SELECTION
1942 027734  012137  002406          MOV     (R1)+,RMTST             ; KEEP ANSWER
1943
1944                                 ; PRINT RUNNING MESSAGE
1945 027740  013701  002514          MOV     UUT,R1
1946 027744                          PRINTF  #RUNNING,R1     ;PRINT RUNNING MESSAGE
1947
1948                                 ;INCREMENT LOGICAL UNIT (DEVICE) NUMBER
1949 027766  005037  002524          CLR     ERRCNT                  ;CLEAN ERROR COUNTER
1950 027772                          EXIT    INIT
1951
1952 027776                          ENDINI:
1953 027776                          ENDINIT
```

```
1954
1955 030000     045     116     045  RUNNING:.ASCIZ  /@N@A RUNNING ON UNIT @D2@A, PASS TIME IS 2 MIN WITHOUT LONG RAM TEST/
1956                                          .EVEN
1957
1958
1959
1960
1961
1962
```

```
                                .SBTTL  AUTODROP SECTION

                                ;**
                                ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                                ; THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                                ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                                ; DROPPED FROM TESTING.
                                ;--
                                .EVEN
030106                                  BGNAUTO

                                .IF NE HELP
                                ;*****************************************************************************
                                ;       INSERT CODE HERE TO CHECK DEVICE(S) TO SEE IF THEY RESPOND.
                                ;       ISSUE A "DODU" FOR THOSE THAT DON'T.
                                ;*****************************************************************************
                                .ENDC


                                ;DEVICE DOES NOT HAVE A "READY"
030106  013701  002474                  MOV     KMVCSR,R1       ;R1 CONTAINS BASE KMV11 ADDRESS
030112  012705  000020                  MOV     #16.,R5         ;16 REGISTERS TO BE TESTED
030116  012737  030150  000004          MOV     #2$,4           ;SET OUT TIMEOUT TRAP
030124  012737  000340  000006          MOV     #340,6          ;LEVEL 7
030132  005711                  1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
030134  000240                          NOP
030136  062701  000002                  ADD     #2,R1           ;NEXT REGISTER
030142  005305                          DEC     R5              ;DEC REGISTER COUNT
030144  001372                          BNE     1$              ;BR IF NOT LAST REGISTER
030146  000405                          BR      3$

030150  062706  000004          2$:     ADD     #4,SP
030154                                  DODU    LOGDEV

030162  013737  002344  000004  3$:     MOV     SAVE4,4
030170  013737  002346  000006          MOV     SAVE6,6
030176                                  ENDAUTO
```

```
2007                            .SBTTL  CLEANUP CODING SECTION
2008
2009                            ;////////////////////////////////////////////////////////////////////////////////
2010                            ;/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
2011                            ;/ AT THE END OF EACH PASS.
2012                            ;////////////////////////////////////////////////////////////////////////////////
2013
2014 030200                             BGNCLN
2015
2016
2017                            .IF NE HELP
2018                            ;*******************************************************************************
2019                            ;       INSERT YOUR CLEANUP CODING.  THIS CODING SHOULD
2020                            ;       RESTORE YOUR TEST-DEVICE TO A NEUTRAL STATE.
2021                            ;       THIS CODE WILL BE EXECUTED AFTER EACH PASS AND AFTER THE
2022                            ;       PROGRAM IS INTERRUPTED BY "↑C".
2023                            ;*******************************************************************************
2024                            .ENDC
2025                            .IF NE HELP
2026                            ;*******************************************************************************
2027                            ;     INSERT LOCAL STORAGE THAT IS USED ONLY
2028                            ;     DURING THE CLEANUP SECTION.
2029                            ;*******************************************************************************
2030
2031                            ;*******************************************************************************
2032                            ;     INSERT MESSAGES THAT ARE USED ONLY
2033                            ;     DURING THE CLEANUP SECTION.
2034                            ;*******************************************************************************
2035                            .ENDC
2036
2037
2038
2039 030200                             BRESET
2040
2041 030202                             ENDCLN
2042
2043
2044
2045
2046
```

```
                        .SBTTL  DROP UNIT SECTION

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

030204                          BGNDU


                        .IF NE HELP
                        ;*******************************************************************************
                        ;       INSERT DROP CODE HERE.  THIS CODE WILL BE EXECUTED AFTER
                        ;       A "DROP" COMMAND OR A "DODU" MACRO EXECUTION.  THE PURPOSE
                        ;       OF THIS CODE IS TO DO ANY NECESSARY HOUSEKEEPING AFTER A
                        ;       UNIT HAS BEEN DROPPED.  THIS SECTION IS OPTIONAL.
                        ;*******************************************************************************
                        .ENDC

                                .EVEN
030204                          PRINTF  @DROPD,R0               ;PRINT "UNIT DROPPED" MESSAGE
030226                          EXIT    DU

030232  045  116  045  DROPD:   .ASCIZ  /%N%A UNIT %D2%A DROPPED/
                                .EVEN


                        .IF NE HELP
                        ;*******************************************************************************
                        ;       INSERT LOCAL STORAGE THAT IS USED ONLY
                        ;       DURING THE DROP-UNIT SECTION.
                        ;*******************************************************************************

                        ;*******************************************************************************
                        ;       INSERT MESSAGES THAT ARE USED ONLY
                        ;       DURING THE DROP-UNIT SECTION.
                        ;*******************************************************************************
                        .ENDC

                                .EVEN


                        ;ISSUE Q-BUS RESET TO CLEAN UP
030262                          BRESET
030264                          ENDDU
```

```
                        .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.
                        ;//////////////////////////////////////////////////////////////////////////////


                        .IF NE HELP
                        ;*****************************************************************************
                        ;       INSERT ADD CODE HERE.  THIS CODE WILL BE EXECUTED AFTER
                        ;       AN "ADD" COMMAND.  THE PURPOSE OF THIS CODE IS TO DO ANY
                        ;       HOUSEKEEPING THAT MAY BE NECESSARY AFTER A UNIT HAS BEEN ADDED.
                        ;       THIS SECTION IS OPTIONAL.
                        ;*****************************************************************************
                        .ENDC

030266                          BGNAU
030266                          ENDAU
```

```
2131
2132                            .IF NE HELP
2133                            ; TEST TO ...
2134
2135                            ;*****************************************************************************
2136                            ;       CHANGE THE PHRASE "TEST TO ..." TO BE A FUNCTIONAL
2137                            ;       DESCRIPTION OF THE HARDWARE TEST WHICH FOLLOWS.
2138                            ;*****************************************************************************
2139                            .ENDC
2140
2141                            .IF NE HELP
2142                            ;*****************************************************************************
2143                            ;       INSERT PROGRAM EQUATES THAT ARE USED ONLY IN THIS TEST.
2144                            ;*****************************************************************************
2145
2146                            ;         BGNTST
2147
2148                            ;*****************************************************************************
2149                            ;       INSERT THE CODING FOR THIS HARDWARE TEST.
2150                            ;*****************************************************************************
2151
2152                            ;         EXIT    TST
2153
2154                            ;*****************************************************************************
2155                            ;       INSERT LOCAL STORAGE THAT IS USED ONLY
2156                            ;       DURING THIS TEST.
2157                            ;*****************************************************************************
2158
2159                            ;*****************************************************************************
2160                            ;       INSERT MESSAGES THAT ARE USED ONLY
2161                            ;       DURING THIS TEST.
2162                            ;*****************************************************************************
2163
2164                            ;         .EVEN
2165                            ;         ENDTST
2166
2167                            ;*****************************************************************************
2168                            ;       BEGIN THE REMAINING TESTS ON NEW PAGES.
2169                            ;*****************************************************************************
2170
2171
2172
2173                            .ENDC
2174
2175 030270                     BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST1 **
                                ;
                                ;
                                ;
                                ;
2176                            ;
2177                            ; TEST Q-BUS ACCESS ON ALL KMV11 CSR'S
```

```
                                        ;
                                        ; DESCRIPTION:  VERIFY THAT REFERENCING KMV11 CSR'S DOES NOT CAUSE A
                                        ;               TIME OUT TRAP
                                        ;
                                        ; REPORT:       ERROR 1         TIME-OUT ERROR
                                        ;--
                                        BGNTST
030270  013705  002474                          MOV     KMVCSR,R5       ;R5 CONTAINS BASE KMV11 ADDRESS
030274  012701  000020                          MOV     #CSRLEN,R1      ;16 REGISTERS TO BE TESTED
030300  012737  030330  000004                  MOV     #2$,4           ;SET OUT TIMEOUT TRAP
030306  012737  000340  000006                  MOV     #340,6          ;LEVEL 7
030314  005715                          1$:     TST     (R5)            ;REFERENCE DEVICE REGISTERS
030316  000240                                  NOP
030320  062705  000002                          ADD     #2,R5           ;NEXT REGISTER
030324  077105                                  SOB     R1,1$           ;LOOP ON ALL CSR REGISTERS
030326  000417                                  BR      3$
030330  062706  000004                  2$:     ADD     #4,SP
030334  010537  002520                          MOV     R5,ADDR
030340  013737  002514  002516                  MOV     UUT,UNIT
030346                                          ERRHRD  0,EM0001,PADFLT ;TIMEOUT ERROR

030356                                          DODU    UUT             ; DROP THIS UNIT /V01.04/
030364                                          DOCLN                   ; END OF SUBPASS /V01.04/
030366  013737  002344  000004  3$:     MOV     SAVE4,4
030374  013737  002346  000006                  MOV     SAVE6,6
030402                                          ESCAPE  TST

030406                                  ENDTST
                                        .EVEN
```

```
2211 030410                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST2 **
                                        ;
                                        ;
                                        ;
                                        ;
2212                                    ;
2213                                    ; TEST THAT ALL CSR'S CAN BE CLEARED
2214                                    ;
2215                                    ; DESCRIPTION:  THE HOST CLEARS ALL CSR'S THEN IT TESTS THEM FOR
2216                                    ;               ALL ZERO.
2217                                    ;
2218                                    ;
2219                                    ; REPORTS:      ERROR 3         ALL CSR'S CAN'T BE CLEARED
2220                                    ;--
2221
2222
2223 030410                             BGNTST
2224 030410  013705  002474                    MOV     KMVCSR,R5               ;GET CSR ADDRESS
2225 030414  012701  002532                    MOV     #REGSPC,R1              ;GET IMAGE SPACE ADDRESS
2226 030420  012737  000020  002470            MOV     #CSRLEN,BUFLEN          ;LOAD LENGTH IN BYTES
2227 030426  004737  011052                    CALL    CLEAR                   ;CLEAR REGSPC BUFFER
2228
2229 030432  012701  002532                    MOV     #REGSPC,R1              ;SOURCE BUFFER ADDRESS
2230 030436  013702  002474                    MOV     KMVCSR,R2               ;DESTINATION BUFFER ADDRESS
2231 030442  004737  011074                    CALL    COPY                    ;COPY REGSPC BUFFER IN CSR
2232
2233 030446  013701  002474                    MOV     KMVCSR,R1               ;GET SOURCE
2234 030452  012702  002572                    MOV     #WRKSPC,R2              ;GET DESTINATION BUFFER ADDRESS
2235 030456  004737  011074                    CALL    COPY                    ;COPY CSR IN WORK SPACE
2236
2237 030462  012701  002532                    MOV     #REGSPC,R1              ;GET SOURCE BUFFER 1
2238 030466  012702  002572                    MOV     #WRKSPC,R2              ;GET SOURCE BUFFER 2
2239 030472  004737  010356                    CALL    COMPAR                  ;COMPARE BOTH
2240 030476  103006                          BCC     1$                      ;NO ERROR
2241
2242 030500                                   ERRHRD  3,EM0003,PRALL
2243 030510                                   ESCAPE  TST
2244
2245 030514  000240                    1$:     NOP
2246 030516                             ENDTST
2247                                    .EVEN
2248
```

```
                        BADHEAD
                        ;**
                        ;
                        ;
                        ;
                        ;                       ** TEST3 **
                        ;
                        ;
                        ;
                        ;
 2251                   ; TEST THAT THE KMV11 CAN BE INITIALIZED
 2252                   ;
 2253                   ;
 2254                   ; DESCRIPTION:  THE HOST SETS SEL0 TO ENTER MAINTENANCE MODE 3
 2255                   ;               AFTER MASTER CLEAR.
 2256                   ;               UPON COMPLETION OF INITIALISATION , THE DCT11
 2257                   ;               MUST CLEAR MCLR THEN ENTER MODE 3(THE DCT11 LOOPS
 2258                   ;               ON ITSELF)
 2259                   ;
 2260                   ; REPORT:       ERROR 2         MASTER CLEAR FAILS TO RESET
 2261                   ;--
 2262
 2263
 2264
 2265 030520           BGNTST
 2266 030520                   MODE 3
 2267 030524           ENDTST
 2268                  .EVEN
 2269
```

```
   2271 030526                          BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                    ** TEST4 **
                                        ;
                                        ;
                                        ;
   2272                                 ;
   2273                                 ; TEST Q-BUS ACCESS ON SEL2 TILL SEL36
   2274                                 ;
   2275                                 ; DESCRIPTION:  THE HOST WRITES PATTERNS IN EACH CSR THEN IT CHECKS
   2276                                 ;               THE SELECTED CSR FOR THE PATTERN AND ALL OTHERS CSR'S
   2277                                 ;               FOR ZERO.ALL CSR'S ARE CLEARED BEFORE
   2278                                 ;               SELECTION OF A NEW CSR FOR TEST.
   2279                                 ;
   2280                                 ; MAINT MODE:   3
   2281                                 ;
   2282                                 ; PATTERNS:     ALL 0
   2283                                 ;               ALL 1
   2284                                 ;               052525
   2285                                 ;               125252
   2286                                 ;               ROTATING 1
   2287                                 ;               ROTATING 0
   2288                                 ;
   2289                                 ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
   2290                                 ;               ERROR 4         DATA COMPARE ERROR ON CSR'S Q-BUS ACCESS
   2291                                 ;--
   2292
   2293
   2294 030526                          BGNTST
   2295 030526  012704  000002                 MOV     #2,R4                   ;START AT OFFSET 2, TO SAVE SEL0
   2296 030532  004737  011116          1$:    CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS IN R5
   2297 030536                                 MODE    3                       ;SELECT MODE 3
   2298 030542  103002                         BCC     1000$
   2299 030544                                 ESCAPE  TST
   2300 030550                          1000$:
   2301 030550  005065  000000                 CLR     SEL0(R5)                ;CLEAN SEL0
   2302
   2303 030554  012701  002532                 MOV     #REGSPC,R1              ;GET SOURCE BUFFER ADDRESS
   2304 030560  004737  011052                 CALL    CLEAR                   ;CLEAR REGSPC IMAGE SPACE
   2305
   2306 030564  060405                         ADD     R4,R5                   ;ADD OFFSET TO CSR
   2307 030566  012703  007204                 MOV     #PATTRN,R3              ;INIT PATTERN TABLE OFFSET
   2308 030572  012337  002360          2$:    MOV     (R3)+,DATA              ;READ CURRENT PATTERN
   2309 030576  013715  002360                 MOV     DATA,(R5)               ;WRITE IT IN THE SELECTED CSR
   2310 030602  013764  002360  002532        MOV     DATA,REGSPC(R4)         ;SAVE IT IN CSR GOOD IMAGE SPACE
   2311
   2312 030610  013701  002474                 MOV     KMVCSR,R1               ;GET SOURCE BUFFER ADDRESS
   2313 030614  012702  002572                 MOV     #WRKSPC,R2              ;GET DESTINATION BUFFER ADDRESS
   2314 030620  004737  011074                 CALL    COPY                    ;COPY CSR IN WORK SPACE
   2315
   2316 030624  012701  002532                 MOV     #REGSPC,R1              ;GET SOURCE BUFFER 1 ADDRESS
   2317 030630  012702  002572                 MOV     #WRKSPC,R2              ;GET SOURCE BUFFER 2 ADDRESS
```

```
2318 030634  004737  010356              CALL    COMPAR          ;COMPARE BOTH
2319 030640  103010                      BCC     3$              ;CORRECT
2320 030642  010437  002404              MOV     R4,NUM1         ;SAVE OFFSET FOR ERROR REPPORT
2321 030646                              ERRHRD  4,EM0004,PRREG  ;IF NOT, REPORT
2322 030656                              ESCAPE  TST
2323
2324 030662  005713              3$:     TST     (R3)            ;ANY-MORE PATTERN?
2325 030664  001342                      BNE     2$              ;YES-LOOP
2326
2327 030666  005724                      TST     (R4)+           ;NEXT OFFSET
2328 030670  022704  000040              CMP     #40,R4          ;ANY-MORE?
2329 030674  003316                      BGT     1$              ;YES-LOOP
2330
2331 030676  000240                      NOP
2332 030700                      ENDTST
2333                             .EVEN
2334
```

```
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST5 **
                                ;
                                ;
                                ;
                                ;
2337                            ;
2338                            ; TEST Q-BUS ACCESS ON SELO CSR
2339                            ;
2340                            ; DESCRIPTION:  THE HOST CLEARS ALL CSR'S THEN WRITES ONE PATTERN
2341                            ;               IN SELO SUCH AS:
2342                            ;                       SELO=PATTERN & 1'S COMP OF MCLR
2343                            ;               THEN THE HOST CHECKS SELO FOR PATTERN & 1'S COMP MCLR
2344                            ;               AND ALL OTHER CSR FOR ZERO.
2345                            ;
2346                            ; MAINT MODE:   3
2347                            ;
2348                            ; PATTERNS:     ALL 0
2349                            ;               ALL 1
2350                            ;               052525
2351                            ;               125252
2352                            ;               ROTATING 0
2353                            ;               ROTATING 1
2354                            ;
2355                            ; REPORTS:      ERROR 2         MASTER FAILS TO RESET
2356                            ;               ERROR 5         DATA COMPARE ERROR ON SELO
2357                            ;--
2358
2359
2360 030702                     BGNTST
2361 030702  004737  011116             CALL    CLRKMV                  ;CLEAR ALL CSR'S AND GET CSR ADDR. IN R5
2362 030706                             MODE    3                       ;SELECT MAINTENANCE MODE 2
2363 030712  103002                     BCC     1000$
2364 030714                             ESCAPE  TST
2365 030720                     1000$:
2366 030720  005065  000000             CLR     SELO(R5)                ;CLEAN SELO
2367
2368 030724  012701  002532             MOV     #REGSPC,R1              ;GET SOURCE BUFFER ADDRESS
2369 030730  004737  011052             CALL    CLEAR                   ;CLEAR REGSPC BUFFER IMAGE SPACE
2370
2371 030734  012703  007204             MOV     #PATTRN,R3              ;INIT PATTERN OFFSET
2372 030740  012337  002360     1$:     MOV     (R3)+,DATA              ;READ CURRENT PATTERN
2373 030744  042737  040000  002360     BIC     #MCLR,DATA              ;CLEAR "MASTER CLEAR " BIT
2374 030752  013737  002360  002532     MOV     DATA,REGSPC             ;SAVE IT IN CSR'S GOOD IMAGE
2375 030760  013765  002360  000000     MOV     DATA,SELO(R5)           ;WRITE IT IN SELO
2376
2377 030766  013701  002474             MOV     KMVCSR,R1               ;GET CSR ADDRESS
2378 030772  012702  002572             MOV     #WRKSPC,R2              ;GET WORK SPACE ADDRESS
2379 030776  004737  011074             CALL    COPY                    ;COPY CSR IN WORK SPACE
2380
2381 031002  012701  002532             MOV     #REGSPC,R1              ;GET SOURCE BUFFER 1 ADDRESS
2382 031006  012702  002572             MOV     #WRKSPC,R2              ;GET SOURCE BUFFER 2 ADDRESS
```

```
2383 031012  004737  010356                CALL    COMPAR                ;COMPARE BOTH
2384 031016  103010                        BCC     2$                    ;CORRECT
2385
2386 031020  005037  002404                CLR     NUM1                  ;GET OFFSET FOR ERROR REPPORT
2387 031024                                ERRHRD  5,EM0004,PRREG        ;NO-REPORT
2388 031034                                ESCAPE  TST
2389
2390 031040  005713                2$:     TST     (R3)                  ;ANY-MORE PATTERN
2391 031042  001356                        BNE     1$                    ;YES-LOOP
2392
2393 031044  000240                        NOP
2394 031046                        ENDTST
2395                               .EVEN
2396
2397
```

```
2399 031050                                BADHEAD
                                           ;**
                                           ;
                                           ;
                                           ;
                                           ;                    ** TEST6 **
                                           ;
                                           ;
                                           ;
2400                                       ;
2401                                       ; TEST DATA TRANSFER THROUGH SELO
2402                                       ;
2403                                       ; DESCRIPTION:  TEST IN TWO PARTS:
2404                                       ;                                THE FIRST TRIES TO ENTER CORRECTLY
2405                                       ;               MAINTENANCE MODE 2(MAINT1)  AFTER  MASTER  CLEAR,
2406                                       ;               GIVEN THE DCT11 THE POSSIBILITY TO RUN TEST ROUTINES.
2407                                       ;               THE HOST SETS SELO WITH MAINTENANCE MODE 2  AND THE
2408                                       ;               PATTERN 252 IN THE LOW BYTE.THE DCT11 MUST READ THAT
2409                                       ;               THEN WRITE BACK SELO WITH MCLR CLEARED AND PATTERN
2410                                       ;               125 IN THE LOW BYTE.
2411                                       ;                                THE SECOND TESTS THAT THE DCT11 CAN
2412                                       ;               READ THE RUN,READ,WRITE BITS AND PATTERN 125 IN THE
2413                                       ;               LOW BYTE.THE DCT11 MUST CLEAR ALL BITS THEN WRITE BACK
2414                                       ;               IN SELO ERROR BIT AND PATTERN 252.
2415                                       ;
2416                                       ; MAINT MODE:   2
2417                                       ;
2418                                       ; KMV ROUTINES: 25 AND 52
2419                                       ;
2420                                       ; REPORTS:      ERROR 6         DATA COMPARE ERROR ON SELO DUE TO THE
2421                                       ;                               IMPOSSIBILITY FOR THE DCT11 TO CORREC-
2422                                       ;                               TLY ACCESS SELO OR TO ENTER MAINT MODE 1
2423                                       ;--
2424
2425
2426 031050                                BGNTST
2427 031050                                BGNSUB
2428 031052  005077  151416                        CLR     @#KMVCSR                                ;CLEAN CSR
2429 031056  012777  051252  151410                MOV     #MCLR!MAINT1!TSTNUM!252,@#KMVCSR        ;SELECT TEST 52 IN KMV
2430 031064                                        WAIT    2,1                                     ;WAIT 1 MS
2431 031074  022777  001125  151372                CMP     #MAINT1!125,@#KMVCSR                    ;CORRECT ANSWER ?
2432 031102  001416                                BEQ     1$                                      ;YES
2433
2434 031104  017737  151364  002352                MOV     @#KMVCSR,BAD                            ;LOAD BAD VALUE OF CSR
2435 031112  012737  001125  002350                MOV     #MAINT1!125,GOOD                        ;LOAD EXPECTED VALUE
2436 031120  005037  002402                        CLR     NUM0                                    ;OFFSET IS 0
2437 031124                                        ERRHRD  6,EM0005,PRSEL
2438 031134                                        ESCAPE  TST
2439
2440 031140                                1$:     DELAY   1
2441 031170                                ENDSUB
2442 031172                                BGNSUB
2443 031174  012777  132125  151272                MOV     #RUN!WRITE!READ!TSTNUM!125,@#KMVCSR     ;SELECT TEST 25 IN KMV
2444 031202                                        WAIT    2,1                                     ;WAIT 1 MS
2445 031212  022777  000652  151254                CMP     #ERR!252,@#KMVCSR                       ;CORRECT ANSWER ?
```

```
2446 031220  001416                          BEQ     1$                      ;YES
2447
2448 031222  017737  151246  002352          MOV     @KMVCSR,BAD             ;LOAD BAD VALUE
2449 031230  012737  000652  002350          MOV     #ERR!252,GOOD           ;LOAD  EXPECTED VALUE
2450 031236  005037  002402                  CLR     NUMO                    ;OFFSET IS 0
2451 031242                                  ERRHRD  6,EM0005,PRSEL
2452 031252                                  ESCAPE  TST
2453
2454 031256  000240                  1$:     NOP
2455 031260                          ENDSUB
2456 031262                                  EXIT    TST
2457
2458 031266                          ENDTST
2459                                 .EVEN
2460
```

```
                                BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST7 **
                                ;
                                ;
                                ;
                                ;
2463                            ;
2464                            ; TEST THAT ALL THE NATIVE FIRMWARE CAN BE ACCESSED
2465                            ;
2466                            ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR CALCULATING AND VERIVYING
2467                            ;               CHECKSUM OF THE NATIVE CODE.THE HOST WAITS FOR 500MS
2468                            ;               AND POLLS SEL0 FOR DCT11 ANSWER:
2469                            ;                       SEL0=TEST NUMBER IF KMV11 CAN'T ANSWER
2470                            ;                       SEL0=0 IF TEST OKE
2471                            ;                       SEL0=100 IF ANY ERROR
2472                            ;                               SEL2 CONTAINS THE OBTAINED CHECKSUM VALUE
2473                            ;
2474                            ; MAINT MODE:   2
2475                            ;
2476                            ; KMV ROUTINE:  01
2477                            ;
2478                            ; REPORTS:      ERROR 11        NO ANSWER FROM THE KMV11
2479                            ;               ERROR 2         MASTER CLEAR FAILS TO RESET
2480                            ;               ERROR 7         CHECKSUM ERROR
2481                            ;--
2482
2483
2484 031270                     BGNTST
2485 031270                             MODE    2               ;SELECT MAINTENANCE MODE 1
2486 031274  103002                     BCC     1000$
2487 031276                             ESCAPE  TST
2488 031302                     1000$:
2489 031302                             TESTNB  1               ;CALL TEST NUMBER 1 IN FW
2490 031310                             WAIT    1,500.          ;WAIT 500 MS AND CHECK SEL0
2491 031320  000422                     BR      1$              ;SEL0 IS 0. CORRECT ANSWER
2492 031322  000406                     BR      2$              ;SEL0 IS 100. ERROR
2493
2494 031324                             ERRHRD  11,EM0011       ;TIMEOUT ERROR
2495 031334                             ESCAPE  TST
2496
2497 031340  013705  002474      2$:    MOV     KMVCSR,R5       ;/V01.01/
2498 031344  016537  000002  002352    MOV     SEL2(R5),BAD    ;/V01.01/ LOAD WRONG CHECKSUM VALUE
2499 031352                             ERRHRD  7,EM0006,PRCHEK ;/V01.01/
2500 031362                             ESCAPE  TST
2501
2502 031366  000240              1$:    NOP
2503 031370                     ENDTST
2504                            .EVEN
```

```
                                BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                          ** TEST8 **
                                ;
                                ;
                                ;
                                ;
2507                            ;
2508                            ; TEST THE NATIVE FIRMWARE REVISION CODE
2509                            ;
2510                            ; DESCRIPTION:   THIS CONTROL IS CARRIED OUT ON FIRST PASS ONLY.
2511                            ;                THE HOST ASKS THE DCT11 FOR THE CURRENT REVISION
2512                            ;                CODE.THEN IT COMPARES TO THE ACTUAL ONE LOADED
2513                            ;                IN THE PROGRAM.
2514                            ;
2515                            ; MAINT MODE:    2
2516                            ;
2517                            ; KMV ROUTINE:   23,24
2518                            ;
2519                            ; REPORTS:       ERROR 11        KMV11 CAN'T ANSWER
2520                            ;                ERROR 2         MASTER CLEAR FAILS TO RESET
2521                            ;                ERROR 10        BAD REVISION CODE
2522                            ;--
2523
2524
2525 031372                    BGNTST
2526 031372  013705  002514            MOV     UUT,R5          ; /V01.04/ IS IT THE FIRST PASS FOR THIS UNIT ?
2527 031376  006305                    ASL     R5              ; /V01.04/
2528 031400  005765  006622            TST     PNTBL(R5)       ; /V01.04/
2529 031404  001070                    BNE     1$              ;EXIT IF NOT
2530 031406  013705  002474            MOV     KMVCSR,R5       ;GET CSR ADDRESS
2531 031412                            MODE    2               ;SELECT MAINTENANCE MODE 1
2532 031416  103002                    BCC     1000$
2533 031420                            ESCAPE  TST
2534 031424                    1000$:
2535 031424                            TESTNB  24              ;TO READ THE ROM BASE ADDRESS
2536 031432                            WAIT    1,2             ;WAIT 2 MS AND CHECK SELO
2537 031442  000407                    BR      2$              ;SELO IS 0, CORRECT
2538 031444  000240                    NOP                     ;NO POSSIBLE ERROR
2539
2540 031446                            ERRHRD  11,EM0011       ;TIMEOUT ERROR
2541 031456                            ESCAPE  TST
2542
2543 031462  016502  000002    2$:     MOV     SEL2(R5),R2     ;READ ROM ADDRESS
2544 031466  062702  000002            ADD     #2,R2           ;TO POINT TO REVISION CODE
2545 031472  010265  000002            MOV     R2,SEL2(R5)     ;SET PARAMETERS FOR MAILBOX ROUTINE
2546 031476  005065  000006            CLR     SEL6(R5)        ;CODE TO READ THE WORD, WHICH ADDRESS IS
2547                                                           ;IN SEL2
2548 031502                            TESTNB  23              ;TO OBTAINED REVISION
2549 031510                            WAIT    1,2             ;WAIT 2 MS AND CHECK SELO
2550 031520  000407                    BR      3$              ;SELO IS 0, CORRECT ANSWER
2551 031522  000240                    NOP                     ;NO POSSIBLE ERROR
2552
```

```
2553 031524                                 ERRHRD  11,EM0011              ;TIMEOUT ERROR
2554 031534                                 ESCAPE  TST
2555
2556 031540                           3$:
2557 031540  016501  000004                  MOV     SEL4(R5),R1            ;READ CODE
2558 031544  023701  002472                  CMP     REVCOD,R1              ;IS THE ACTUAL REVISION?
2559 031550  001406                          BEQ     1$                     ;YES
2560
2561 031552                                 ERRHRD  10,EM0007,PRREV        ;NO, REPORT
2562 031562                                 ESCAPE  TST
2563
2564 031566  000240                    1$:   NOP
2565 031570                           ENDTST
2566                                  .EVEN
2567
```

```
 2569 031572                    BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST9 **
                                ;
                                ;
                                ;
 2570                           ; TEST DATA TRANSFER THROUGH REGISTER SEL2
 2571                           ;
 2572                           ; DESCRIPTION:   THE HOST WRITES ONE PATTERN IN SEL2, WHICH
 2573                           ;               MUST BE READ BY THE DCT11. THE DCT11 ANSWERS
 2574                           ;               BY LOADING IN SELO THE CORRESPONDING RANGE
 2575                           ;               OF THE PATTERN IT HAS READ.
 2576                           ;
 2577                           ;                       PATTERN RANGE   FOR     PATTERN
 2578                           ;               SELO =  0                       ALL 0
 2579                           ;                       1                       ALL 1
 2580                           ;                       2                       052525
 2581                           ;                       3                       125252
 2582                           ;                       4 - 19.                 ROTATING 0
 2583                           ;                       20. - 35.               ROTATING 1
 2584                           ;                       36.                     UNKNOWN
 2585                           ;               THEN THE HOST TESTS THE CODE RETURNED.
 2586                           ;
 2587                           ; MAINT MODE:   2
 2588                           ;
 2589                           ; KMV ROUTINE:  03
 2590                           ;
 2591                           ; REPORTS:      ERROR 11        NO ANSWER FROM THE KMV11
 2592                           ;               ERROR 2         MASTER CLEAR FAILS TO RESET
 2593                           ;               ERROR 12        DCT11 CAN'T CORRECTLY READ
 2594                           ;                               SEL2
 2595                           ;
 2596                           ;--
 2597
 2598 031572                    BGNTST
 2599 031572  013705  002474            MOV     KMVCSR,R5               ;GET CSR ADDRESS
 2600 031576                            MODE    2                       ;SELECT MAINTENANCE MODE 1
 2601 031602  103002                    BCC     1000$
 2602 031604                            ESCAPE  TST
 2603 031610                    1000$:
 2604 031610  012702  007204            MOV     #PATTRN,R2              ;POINT TO PATTERN TABLE
 2605 031614                    1$:
 2606 031614  012237  002360            MOV     (R2)+,DATA              ;READ CURRENT PATTERN
 2607 031620  013765  002360  000002    MOV     DATA,SEL2(R5)           ;WRITE IT IN SEL2
 2608 031626                            TESTNB  3                       ;SELECT TEST03 IN FW
 2609 031634                            WAIT    2,2                     ;WAIT FOR 2 MS
 2610 031644  116501  000000            MOVB    SELO(R5),R1             ;GET SELO
 2611 031650  020127  000044            CMP     R1,#36.                 ;IS DATA RETURN CORRECT?
 2612 031654  103412                    BLO     2$                      ;YES, CORRECT.
 2613
 2614 031656  010137  002352            MOV     R1,BAD                  ;IF NOT, SET BAD
 2615 031662  005037  002402            CLR     NUMO                    ;POINT TO SELO
```

```
2616 031666                                    ERRHRD  12,EM0010,PRBAD         ;REPORT
2617 031676                                    ESCAPE  TST
2618
2619 031702  006301                      2$:   ASL     R1                      ;CREATE OFFSET
2620 031704  013737  002360  002350            MOV     DATA,GOOD               ;SET GOOD
2621 031712  012737  000002  002402            MOV     #2,NUM0                 ;POINT TO SEL2
2622 031720  016137  007204  002352            MOV     PATTRN(R1),BAD          ;READ RESULT IN BAD
2623 031726  023737  002352  002350            CMP     BAD,GOOD                ;THE SAME?
2624 031734  001406                            BEQ     3$                      ;IF YES
2625
2626 031736                                    ERRHRD  12,EM0010,PRSEL         ;IF NOT, REPORT
2627 031746                                    ESCAPE TST
2628
2629 031752  005712                      3$:   TST     (R2)                    ;ANY-MORE PATTERN?
2630 031754  001317                            BNE     1$                      ;YES, LOOP
2631
2632 031756                              ENDTST
2633                                     .EVEN
```

```
2635 031760                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST10 **
                                        ;
                                        ;
                                        ;
                                        ;
2636                                    ; TEST DATA TRANSFER THROUGH ALL CSR'S EXCEPT SEL0
2637                                    ;
2638                                    ; DESCRIPTION:   THE HOST WRITES ONE PATTERN IN ONE CSR AND
2639                                    ;                ASKS THE DCT11 FOR READING THEN WRITING IT
2640                                    ;                IN ANOTHER CSR.THE DCT11 INDICATES COMPLE-
2641                                    ;                TION OF THE TRANSFER BY CLEARING SEL0. THEN
2642                                    ;                THE HOST TESTS THE SELECTED CSR'S FOR PAT-
2643                                    ;                TERN AND ALL OTHER CSR'S FOR ZERO.
2644                                    ;                ALL CSR ARE CLEARED BEFORE SELECTION OF NEW
2645                                    ;                CSR'S FOR TEST.
2646                                    ;                THE HOST USES SEL2 TO PASS THE SOURCE AND
2647                                    ;                DESTINATION CSR OFFSETS IN THE FOLLOWING WAY.
2648                                    ;
2649                                    ;                        15              8 7              0
2650                                    ;                        ---------------------------------
2651                                    ;                SEL2    )    SOURCE     ) DESTINATION  )
2652                                    ;                        ---------------------------------
2653                                    ;
2654                                    ;                THE TEST INCLUDES THE FOLLOWING TRANSFERS:
2655                                    ;                        FROM             TO
2656                                    ;
2657                                    ;                        SEL4             SEL2
2658                                    ;                        SEL4             SEL6
2659                                    ;                        SEL6             SEL10
2660                                    ;                        SEL10            SEL12
2661                                    ;                        SEL12            SEL14
2662                                    ;                        SEL14            SEL16
2663                                    ;                        SEL16            SEL20
2664                                    ;                        SEL20            SEL22
2665                                    ;                        SEL22            SEL24
2666                                    ;                        SEL24            SEL26
2667                                    ;                        SEL26            SEL30
2668                                    ;                        SEL30            SEL32
2669                                    ;                        SEL32            SEL34
2670                                    ;                        SEL34            SEL36
2671                                    ;                        SEL36            SEL4
2672                                    ;
2673                                    ; MAINT MODE:    2
2674                                    ;
2675                                    ; KMV ROUTINE:   04
2676                                    ;
2677                                    ; REPORTS:       ERROR 2          MASTER CLEAR FAILS TO RESET
2678                                    ;                ERROR 11         KMV11 CAN'T ANSWER
2679                                    ;                ERROR 13         Q-BUS CAN'T CORRECTLY ACCESS
2680                                    ;                                 CSR'S
2681                                    ;
```

```
                                        ;--
                                        BGNTST
031760                                  
                                        MOV     @REGTAB,R4              ;POINT TO REGISTER TRANSFER TABLE
031760  012704  007316                  
                                        1$:
031764                                  
031764  012701  002532                  MOV     @REGSPC,R1              ;POINT TO CSR GOOD IMAGE
031770  012737  000020  002470          MOV     @CSRLEN,BUFLEN          ;LOAD LENGTH IN BYTES
031776  004737  011052                  CALL    CLEAR                   ;CLEAN IMAGE SPACE
032002  004737  011116                  CALL    CLRKMV                  ;CLEAN CSR'S
032006                                  MODE    2                       ;SELECT MAINTENANCE MODE 1
032012  103002                          BCC     1000$
032014                                  ESCAPE  TST
032020                                  1000$:
032020  012703  007204                  MOV     @PATTRN,R3              ;POINT TO PATTERN TABLE
032024                                  2$:
032024  111401                          MOVB    (R4),R1                 ;GET "FROM" POINTER
032026  001474                          BEQ     5$                      ;QUIT IF NO MORE ENTRY IN REGTAB TABLE
032030  116402  000001                  MOVB    1(R4),R2                ;GET "TO" POINTER
032034  012337  002360                  MOV     (R3)+,DATA              ;READ CURRENT PATTERN
032040  110165  000003                  MOVB    R1,SEL3(R5)             ;LOAD SOURCE OFFSET IN CSR
032044  110265  000002                  MOVB    R2,SEL2(R5)             ;LOAD DESTINATION OFFSET IN CSR
032050  010546                          MOV     R5,-(SP)                ;SAVE CSR ADDRESS
032052  060116                          ADD     R1,(SP)                 ;GET SELECTED CSR ADDRESS IN SP
032054  013736  002360                  MOV     DATA,@(SP)+             ;LOAD THE PATTERN IN THE SELECTED CSR
032060  013761  002360  002532          MOV     DATA,REGSPC(R1)         ;LOAD PATTERN IN IMAGE SPACE AT SOURCE OFFSET
032066  013762  002360  002532          MOV     DATA,REGSPC(R2)         ;LOAD PATTERN IN IMAGE SPACE AT DEST. OFFSET
032074                                  TESTNB  4                       ;CALL TEST04 IN FW

032102                                  WAIT    1,2                     ;WAIT 2 MS AND CHECK SEL0
032112  000407                          BR      3$                      ;SEL0=0, CORRECT ANSWER
032114  000240                          NOP                             ;NO POSSIBLE ERROR
032116                                  ERRHRD  11,EM0011               ;TIMEOUT ERROR
032126                                  ESCAPE  TST

032132  013701  002474          3$:     MOV     KMVCSR,R1               ;GET SOURCE BUFFER ADDRESS = CSR
032136  012702  002572                  MOV     @WRKSPC,R2              ;GET DESTINATION BUFFER ADDRESS
032142  004737  011074                  CALL    COPY                    ;COPY CSR IN WORK SPACE

032146  012701  002532                  MOV     @REGSPC,R1              ;GET BUFFER 1 ADDRESS
032152  012702  002572                  MOV     @WRKSPC,R2              ;GET BUFFER 2 ADDRESS
032156  004737  010356                  CALL    COMPAR                  ;COMPARE WRKSPC BUFFER TO REGSPC BUFFER
032162  103011                          BCC     4$                      ;IF THE SAME

032164  116437  000001  002404          MOVB    1(R4),NUM1              ;LOAD "TO" OFFSET FOR ERROR MESSAGE
032172                                  ERRHRD  13,EM0012,PRREG         ;IF NOT, REPORT
032202                                  ESCAPE  TST

032206  005713                  4$:     TST     (R3)                    ;ANY-MORE PATTERN?
032210  001305                          BNE     2$                      ;IF YES, LOOP

032212                                  BREAK
032214  005724                          TST     (R4)+                   ;NEXT ENTRY IN REGISTER TABLE
032216  000662                          BR      1$                      ;CONTINUE
032220  000240                  5$:     NOP
```

```
2739 032222                    ENDTST
2740                           .EVEN
```

```
                                BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST11 **
                                ;
                                ;
                                ;
                                ;
2743                            ;  KMV11-C EPROM'S PART NUMBER PRINT OUT
2744                            ;
2745                            ; DESCRIPTION:  THIS CONTROL IS CARRIED  OUT ON  FIRST  PASS
2746                            ;               ONLY.THE GOAL IS TO GET THE PART NUMBER STO-
2747                            ;               RED IN THE KMV EPROM THEN TO PRINT IT.
2748                            ;               IN ORDER TO DO THAT, THE HOST ASKS THE DCT11
2749                            ;               TO RETURN BACK THE ROM BASE  ADDRESS, SUCH
2750                            ;               AS:
2751                            ;                       SEL0 = 0 FOR KMV ROUTINE 24 JOB DONE
2752                            ;                       SEL2 = ROM BASE ADDRESS
2753                            ;               THEN THE HOST BY MEANS OF THE  KMV 'MAILBOX'
2754                            ;               ROUTINE READS ALL PART NUMBER CARACTERS. FOR
2755                            ;               THIS ROUTINE THE HOST PASSES PARAMETERS:
2756                            ;                       SEL2 = PART NUMBER ADDRESS
2757                            ;                       SEL4 = BYTE READ
2758                            ;                       SEL6 = 4 FOR READ BYTE INDICATION
2759                            ;               THEN THE HOST PRINTS PART NUMBER.
2760                            ;
2761                            ; MAINT MODE:   2
2762                            ;
2763                            ; KMV ROUTINES: 23,24
2764                            ;
2765                            ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
2766                            ;               ERROR 11        NO KMV11 RESPONSE
2767                            ;
2768                            ;--
2769
2770
2771 032224                     BGNTST
2772 032224  013701  002514            MOV     UUT,R1                  ; /V01.04/ IS IT THE FIRST PASS FOR THIS UNIT ?
2773 032230  006301                    ASL     R1
2774 032232  005761  006622            TST     PNTBL(R1)               ; /V01.04/
2775 032236  001402                    BEQ     5$                      ;CONTINUE IF NOT
2776 032240                            EXIT    TST                     ;IF NOT, EXIT
2777 032244                     5$:
2778 032244  004737  011116            CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS IN R5
2779 032250                            MODE    2                       ;SELECT MAINTENANCE MODE 1
2780 032254  103002                    BCC     1000$
2781 032256                            ESCAPE  TST
2782
2783 032262                     1000$:
2784 032262  013701  002514            MOV     UUT,R1                  ; /V01.04/
2785 032266  006301                    ASL     R1                      ; /V01.04/
2786 032270  005261  006622            INC     PNTBL(R1)               ; /V01.04/ MARK THAT THE FIRST PASS IS DONE
2787 032274                            TESTNB  24                      ;SELECT TEST 24 IN FW
2788 032302                            WAIT    1,2                     ;WAIT 2 MS AND CHECK SEL0
```

```
2789 032312  000407                        BR      1$                      ;SELO IS 0, CORRECT
2790 032314  000240                        NOP                             ;NO POSSIBLE ERROR
2791 032316                                ERRHRD  11,EM0011               ;REPORT TIMEOUT ERROR
2792 032326                                ESCAPE  TST
2793
2794 032332                          1$:
2795 032332  012702  006764                MOV     #PARTNB,R2              ;PART NUMBER BUFFER ADD.
2796 032336  016503  000002                MOV     SEL2(R5),R3             ;GET ROM BASE ADD.
2797 032342  062703  000004                ADD     #4,R3                   ;POINT TO PART NUMBER
2798 032346  012765  000004  000006        MOV     #4,SEL6(R5)             ;SET FLAG FOR BYTE READ
2799 032354                          2$:
2800 032354  010365  000002                MOV     R3,SEL2(R5)             ;LOAD PART NUMBER ADDRESS
2801 032360                                TESTNB  23                      ;SELECT TEST 23 IN FW
2802 032366                                WAIT    1,2                     ;WAIT 2MS AND CHECK SELO
2803 032376  000407                        BR      3$                      ;SELO IS 0, CORRECT
2804 032400  000240                        NOP                             ;NO POSSIBLE ERROR
2805 032402                                ERRHRD  11,EM0011               ;REPORT TIMEOUT ERROR
2806 032412                                ESCAPE  TST
2807
2808 032416                          3$:
2809 032416  116522  000004                MOVB    SEL4(R5),(R2)+          ;GET BYTE AND LOAD IT IN PART NB BUFFER
2810 032422  005203                        INC     R3                      ;NEXT BYTE
2811 032424  020227  007030                CMP     R2,#PARTNB+36.          ;BOTTOM OF BUFFER? /V01.04/
2812 032430  002751                        BLT     2$                      ;NOT, LOOP
2813
2814 032432  012737  006764  002350        MOV     #PARTNB,GOOD            ;LOAD LOW ROM PART NUMBER ADDRESS
2815 032440                                PRINTF  #MPNUML,GOOD            ;PRINT OUT LOW PART NUMBER
2816 032464  012737  007006  002350        MOV     #PARTNB+18.,GOOD        ;LOAD HIGH ROM PART NUMBER ADDRESS /V01.04/
2817 032472                                PRINTF  #MPNUMH,GOOD            ;PRINT OUT HIGH PART NUMBER
2818
2819 032516                                ENDTST
2820                                       .EVEN
2821
```

```
2823 032520                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST12 **
                                        ;
                                        ;
                                        ;
                                        ;
2824                                    ; SHORT RAM TEST
2825                                    ;
2826                                    ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING THE RAM
2827                                    ;               BY ITSELF. PERIODICALLY THE HOST CHECKS THE
2828                                    ;               REGISTER SEL0 FOR ANY DCT11 RESPONSE:
2829                                    ;                       SEL0 = 0        IF TEST SUCCEEDED
2830                                    ;                       SEL0 = 100      IF ANY ERROR
2831                                    ;               IN THE LATTER CASE, THE FOLLOWING REGISTERS
2832                                    ;               CONTAINS THE ERROR REPORTS AS FOLLOW:
2833                                    ;
2834                                    ;               - IF THE DCT11 CAN'T CLEAR ALL THE RAM
2835                                    ;                       SEL2 = 2
2836                                    ;                       SEL4 = FIRST BAD WORD ADDRESS
2837                                    ;                       SEL6 = FIRST BAD WORD
2838                                    ;
2839                                    ;               - IF DATA COMPARE ERROR AT LOCATION TESTED
2840                                    ;                       SEL2 = 4
2841                                    ;                       SEL4 = RAM ADDRESS
2842                                    ;                       SEL6 = BAD WORD
2843                                    ;                       SEL10= EXPECTED VALUE
2844                                    ;
2845                                    ;
2846                                    ; MAINT MODE:   2
2847                                    ;
2848                                    ; KMV ROUTINE:  26
2849                                    ;
2850                                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
2851                                    ;               ERROR 11        NO ANSWER FROM THE KMV11
2852                                    ;               ERROR 14        ALL THE RAM CAN'T BE CLEARED
2853                                    ;               ERROR 15        DATA COMPARE ERROR ON RAM AT
2854                                    ;                               LOCATION UNDER TEST
2855                                    ;
2856                                    ;--
2857
2858
2859 032520                             BGNTST
2860 032520  013705  002474                    MOV     KMVCSR,R5               ;GET CSR ADDRESS
2861 032524                                    MODE    2                       ;SELECT MAINTENANCE MODE 1
2862 032530  103002                            BCC     1000$
2863 032532                                    ESCAPE  TST
2864 032536                             1000$:
2865 032536                                    TESTNB  26                      ;SELECT TEST 26 IN FW
2866
2867 032544                                    WAIT    1,10000.                ;WAIT AND CHECK SEL0
2868 032554  000451                            BR      6$                      ;SEL0 IS 0, CORRECT
2869 032556  000406                            BR      2$                      ;IF ANY ERROR
```

```
2870
2871 032560                                    ERRHRD  11,EM0011              ;IF NO RESPONSE
2872 032570                                    ESCAPE TST
2873
2874 032574                              2$:   DSPACM  3
2875 032602  000403                            BR      3$                     ;FOR ERROR 14
2876 032604  000416                            BR      4$                     ;FOR ERROR 15
2877
2878 032606                                    ESCAPE  TST
2879
2880 032612  016537  000004  002414  3$:     MOV     SEL4(R5),BADLOC        ;GET BAD RAM ADDRESS
2881 032620  016537  000006  002352            MOV     SEL6(R5),BAD           ;GET BAD VALUE
2882 032626                                    ERRHRD  14,EM0013,PRRAM1       ;REPORT
2883 032636                                    ESCAPE  TST
2884
2885 032642  016537  000004  002416  4$:     MOV     SEL4(R5),TSTLOC        ;GET SELECTED RAM ADDRESS
2886 032650  016537  000006  002412            MOV     SEL6(R5),BDDAT         ;GET BAD VALUE
2887 032656  016537  000010  002360            MOV     SEL10(R5),DATA         ;GET EXPECTED VALUE
2888 032664                                    ERRHRD  15,EM0014,PRRAM2       ;REPORT
2889 032674                                    ESCAPE  TST
2890
2891
2892 032700  000240                      6$:   NOP
2893 032702                              ENDTST
2894                                     .EVEN
2895
```

```
2897 032704                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST13 **
                                        ;
                                        ;
                                        ;
2898                                    ; TEST OF INTERNAL KMV11 R/W REGISTERS
2899                                    ;
2900                                    ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING ALL R/W
2901                                    ;               REGISTERS BY ITSELF. PERIODICALLY, THE HOST
2902                                    ;               POLLS SEL0 FOR ANY DCT11 RESPONSE:
2903                                    ;                       SEL0 = 0        FOR TEST OKE
2904                                    ;                       SEL0 = 100      FOR ANY ERROR
2905                                    ;               IN THE LATTER CASE, THE FOLLOWING REGISTERS
2906                                    ;               CONTAINS ERROR REPORTS, SUCH AS:
2907                                    ;                       SEL2 = 2        BYTE COUNT OUT REG.
2908                                    ;                            = 4        BUS ADDRESS OUT REG.
2909                                    ;                            = 6        EXT ADDRESS OUT REG.
2910                                    ;                            = 10       BYTE COUNT IN REG.
2911                                    ;                            = 12       BUS ADDRESS IN REG.
2912                                    ;                            = 14       EXT ADDRESS IN REG.
2913                                    ;                       SEL4 = CURRENT PATTERN
2914                                    ;                       SEL6 = BAD WORD
2915                                    ;
2916                                    ; MAINT MODE:   2
2917                                    ;
2918                                    ; KMV ROUTINE:  06
2919                                    ;
2920                                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
2921                                    ;               ERROR 11        NO RESPONSE FROM THE KMV11
2922                                    ;               ERROR 17        DATA COMPARE ERROR ON KMV11
2923                                    ;                               INTERNAL R/W REGISTER
2924                                    ;
2925                                    ;
2926                                    ;--
2927
2928
2929 032704                             BGNTST
2930 032704  004737  011116                        CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS
2931 032710                                        MODE    2                       ;SELECT MAINTENANCE MODE 1
2932 032714  103002                               BCC     1000$
2933 032716                                        ESCAPE  TST
2934 032722                             1000$:
2935 032722                                        TESTNB  6                       ;SELECT TEST 6 IN FW
2936 032730                                        WAIT    1,100.                  ;WAIT 100MS AND CHECK SEL0
2937 032740  000426                               BR      3$                      ;SEL0 IS 0, CORRECT
2938 032742  000406                               BR      1$                      ;IF ANY ERROR
2939
2940 032744                                        ERRHRD  11,EM0011               ;IF NO RESPONSE
2941 032754                                        ESCAPE TST
2942
2943 032760  016537  000002  002352  1$:         MOV     SEL2(R5),BAD            ;GET ERROR CODE
```

```
 2944 032766  016537  000004  002360          MOV     SEL4(R5),DATA           ;GET PATTERN
 2945 032774  016537  000006  002412          MOV     SEL6(R5),BDDAT          ;GET BAD VALUE
 2946 033002                                  ERRHRD  17,EM0016,INREG         ;REPORT
 2947 033012                                  ESCAPE TST
 2948
 2949 033016  000240                  3$:     NOP
 2950 033020                          ENDTST
 2951                                 .EVEN
 2952
 2953
```

```
2955 033022                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST14 **
                                        ;
                                        ;
                                        ;
2956                                    ; TEST INTERRUPT CAPABILITY OF KMV11-C ON Q-BUS
2957                                    ;
2958                                    ; DESCRIPTION:  THE HOST GETS ACTUAL VECTOR AND PRIORITY
2959                                    ;               FROM P.TABLE. THE HOST PREPARES VECTOR AREA
2960                                    ;               AND SETS ITS PRIORITY TO 7 THEN IT ASKS THE
2961                                    ;               DCT11 FOR REQUESTING AN INTERRUPT ON Q-BUS.
2962                                    ;               SINCE THE DCT11 CAN REQUEST INTERRUPT ON
2963                                    ;               Q-BUS THROUGH VECTORS XXO AND XX4 THE TEST
2964                                    ;               IS COMPOSED OF TWO SUBTESTS, ONE FOR EACH
2965                                    ;               VECTOR. THE HOST INDICATES TO THE DCT11 THE
2966                                    ;               VECTOR UNDER TEST BY MEANS OF SEL2:
2967                                    ;                       SEL2 = 0        FOR XXO
2968                                    ;                       SEL2 = 1        FOR XX4
2969                                    ;               THE DCT11 RESPONSE AFTER COMPLETION IS:
2970                                    ;                       SELO = 0
2971                                    ;               THEN,
2972                                    ;               FOR LEVEL=6 TO 3 DO
2973                                    ;                       NOP
2974                                    ;                       NOP
2975                                    ;                       READ INTERRUPT FLAG
2976                                    ;                       IF FLAG SET THEN
2977                                    ;                               IF AT LEGAL PRIORITY THEN
2978                                    ;                                       GOOD EXIT
2979                                    ;                               ELSE
2980                                    ;                                       ERROR REPORTS
2981                                    ;                               ENDIF
2982                                    ;                       ENDIF
2983                                    ;               ENDDO
2984                                    ;               REPORT ERROR FOR NO INTERRUPT
2985                                    ;
2986                                    ; MAINT MODE:   2
2987                                    ;
2988                                    ; KMV ROUTINE:  07
2989                                    ;
2990                                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
2991                                    ;               ERROR 11        NO KMV11 RESPONSE
2992                                    ;               ERROR 20        Q-BUS INTERRUPT AT AN UNEXPECTED PRIORITY LEVEL (NOT USED)
2993                                    ;               ERROR 21        NO Q-BUS INTERRUPT OCCURS
2994                                    ;
2995                                    ;--
2996
2997
2998 033022                             BGNTST
2999 033022                             BGNSUB
3000 033024  013705  002474                    MOV     KMVCSR,R5               ;GET CSR ADDRESS
3001 033030                                    MODE    2                       ;SELECT MAINTENANCE MODE 1
```

```
3002 033034  103002                          BCC     1000$
3003 033036                                  ESCAPE  TST
3004 033042                          1000$:
3005 033042  005065  000002                  CLR     SEL2(R5)            ;FOR VECTOR XX0
3006 033046  013701  002476                  MOV     KMVV00,R1           ;SET VECTOR XX0 ADDRESS
3007 033052  004737  033130                  CALL    QBUSIT              ;RUN TEST
3008 033056                          ENDSUB
3009
3010 033060                          BGNSUB
3011 033062  013705  002474                  MOV     KMVCSR,R5           ;GET CSR ADDRESS
3012 033066                                  MODE    2                   ;SELECT MAINTENANCE MODE 1
3013 033072  103002                          BCC     1000$
3014 033074                                  ESCAPE  TST
3015 033100                          1000$:
3016 033100  012765  000001  000002          MOV     #1,SEL2(R5)         ;FOR VECTOR XX4
3017 033106  013701  002500                  MOV     KMVV04,R1           ;SET VECTOR XX4 ADDRESS
3018 033112  004737  033130                  CALL    QBUSIT              ;RUN TEST
3019 033116                          ENDSUB
3020
3021 033120  012703  000340                  MOV     #PRI07,R3           ; RESTORE HOST PRIORITY /V01.04/
3022 033124  106403                          MTPS    R3                  ; /V01.04/
3023
3024
3025 033126                          ENDTST
3026                                 .EVEN
3027
3028                                 ;
3029                                 ;       Q-BUS INTERRUPT TEST ROUTINE
3030                                 ;
3031
3032 033130                          QBUSIT:
3033 033130  005037  002356                  CLR     INTFLG              ;RESET INT. FLAG
3034 033134  010137  002424                  MOV     R1,GDVEC            ;SAVE VECTOR
3035 033140  012721  033304                  MOV     #T14A00,(R1)+       ;LOAD ADDRESS IN  VECTOR
3036 033144  012711  000340                  MOV     #PRI07,(R1)         ;LOAD PRIORITY IN 2ND WORD
3037 033150  013737  002502  002422          MOV     KMVLVL,GDLVL        ;SAVE GOOD LEVEL
3038 033156                                  DO      <ASR    GDLVL>,4    ;GET PRIORITY LEVEL RIGTH JUSTIFIED
3039 033176  012703  000340                  MOV     #PRI07,R3           ;INIT PRIORITY LEVEL TO 7
3040 033202  106403                          MTPS    R3                  ;SET HOST PRIORITY
3041 033204                                  TESTNB  7                   ;SELECT TEST07 IN FW
3042 033212                                  WAIT    1,1                 ;WAIT 1 MS AND CHECK SEL0
3043 033222  000406                          BR      1$                  ;SEL0 IS 0, CORRECT ANSWER
3044 033224  000240                          NOP                         ;NO POSSIBLE ERROR
3045 033226                                  ERRHRD  11,EM0011           ;IF NO RESPONSE
3046 033236  000207                          RETURN
3047
3048 033240  162703  000040          1$:     SUB     #40,R3              ;DEC PRIORITY LEVEL OF ONE
3049 033244  022703  000100                  CMP     #PRI02,R3           ;LOWEST REACHED ?
3050 033250  001407                          BEQ     2$                  ;IF YES
3051 033252  106403                          MTPS    R3                  ;IF NOT, SET NEW HOST PRIORITY
3052 033254                                  WAIT    0,1                 ;WAIT 1 MS AND CHECK IF INTERRUPT OCCURS
3053 033264  000406                          BR      3$                  ;IF ANY INTERRUPT
3054 033266  000764                          BR      1$                  ;IF TIME-OUT
3055
3056 033270                          2$:     ERRHRD  21,EM0020,PRVEC     ;NO INTERRUPT OCCURS
3057 033300  000207                          RETURN
3058
```

```
3059 033302                        3$:
3060                               ;       INT LEVEL NOT CHECKED ON 11/23 /V01.03/
3061                               ;
3062                               ;       ADD     #40,R3                  ;ADJUST LEVEL
3063                               ;       MOV     R3,BDLVL                ;SAVE IT
3064                               ;       CMP     BDLVL,GDLVL             ;IS AT THE GOOD ONE?
3065                               ;       BEQ     4$                      ;IF YES
3066                               ;       DO      <ASR    BDLVL>,5        ;GET CURRENT PRIORITY RIGHT JUSTIFIED
3067                               ;       DO      <ASR    GDLVL>,5
3068                               ;       ERRHRD  20,EM0017,PRLVL         ;UNEXPECTED PRIORITY LEVEL
3069
3070 033302  000207                4$:     RETURN
3071
3072
3073                               ;
3074                               ;       TEST14 INTERRUPT ROUTINE
3075                               ;
3076
3077 033304                        T14A00:
3078 033304  005237  002356                INC     INTFLG                  ;SET INT. FLAG
3079 033310  000002                        RTI
3080
3081
```

```
   3083 033312                          BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST15 **
                                        ;
                                        ;
                                        ;
                                        ;
   3084                                 ; TEST THAT ACCESS TO BSEL0/BSEL2 INTERRUPTS THE DCT11
   3085                                 ;
   3086                                 ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR SERVICING INTER-
   3087                                 ;               RUPT ON BSEL0/BSEL2 INTERRUPT. SINCE BSEL0
   3088                                 ;               AND BSEL2 ACCESSES CAN PRODUCE AN INTERRUPT,
   3089                                 ;               THE HOST USES SEL2 TO SPECIFY TO THE DCT11
   3090                                 ;               WHICH CSR ACCESS IS TESTED:
   3091                                 ;                       SEL2 = 0        FOR BSEL0 ACCESS
   3092                                 ;                       SEL2 = 2        FOR BSEL2 ACCESS
   3093                                 ;               DCT11 RESPONSE WILL BE:
   3094                                 ;                       SEL0 = 0     IF EXPECTED INTERRUPT
   3095                                 ;                            = 100   IF INTERRUPT AT ILLEGAL VECTOR
   3096                                 ;                            = OTHER IF NOT INTERRUPTED
   3097                                 ;               THEN AFTER ASKING THE DCT11,
   3098                                 ;
   3099                                 ;               DO FOR BSEL=0 TO 37 EXCEPT BSEL0/2
   3100                                 ;                       ACCESS ON CSR
   3101                                 ;                       WAIT FOR 1 MS
   3102                                 ;                       READ SEL0
   3103                                 ;                       CASE SEL0 OF
   3104                                 ;                               0 : REPORT ERROR 23 AND EXIT
   3105                                 ;                             100 : REPORT ERROR 23 AND EXIT
   3106                                 ;                       ENDCASE
   3107                                 ;               ENDDO
   3108                                 ;               SET BITS IN BSEL0/2
   3109                                 ;               WAIT FOR 1 MS
   3110                                 ;               READ SEL0
   3111                                 ;               IF SEL0=0 THEN
   3112                                 ;                       GOOD EXIT
   3113                                 ;               IF SEL0=100 THEN
   3114                                 ;                       REPORT ERROR 24 AND EXIT
   3115                                 ;               ELSE
   3116                                 ;                       IF SEL4=2 THEN
   3117                                 ;                               REPORT ERROR 22 AND EXIT
   3118                                 ;                       ELSE
   3119                                 ;                               REPORT ERROR 11 AND EXIT
   3120                                 ;                       ENDIF
   3121                                 ;               ENDIF
   3122                                 ;
   3123                                 ; MAINT MODE:   2
   3124                                 ;
   3125                                 ; KMV ROUTINE:  10
   3126                                 ;
   3127                                 ; REPORTS:      ERROR 2         MASTERCLEAR FAILS TO RESET
   3128                                 ;               ERROR 11        NO KMV11 RESPONSE
   3129                                 ;               ERROR 22        DCT11 NOT INTERRUPTED
```

```
                        ;               ERROR 23        UNEXPECTED KMV11 INTERRUPT ON CSR ACCESS
                        ;               ERROR 24        KMV11 INTERRUPTED AT ILLEGAL VECTOR
                        ;
                        ;--

                        BGNTST
                        BGNSUB
                                MODE    2
                                BCC     1000$
                                ESCAPE  TST
                        1000$:
                                CLR     R1                      ;SET FLAG FOR BSELO TEST
                                CALL    BSELIT                  ;RUN TEST
                        ENDSUB

                        BGNSUB
                                MODE    2
                                BCC     1000$
                                ESCAPE  TST
                        1000$:
                                MOV     #2,R1                   ;SET FLAG FOR BSEL2 TEST
                                CALL    BSELIT                  ;RUN TEST
                        ENDSUB

                                EXIT    TST

                        ;
                        ;       TEST15 COMMON TEST PART
                        ;

033370                  BSELIT:
033370  013705  002474          MOV     KMVCSR,R5               ;GET CSR ADDRESS
033374  010137  002402          MOV     R1,NUMO                 ;SAVE CSR OFFSET FOR ERROR MESSAGE
033400  010165  000002          MOV     R1,SEL2(R5)             ;INFORM DCT11 OF THE SELECTED OFFSET
033404                          TESTNB  10                      ;SELECT TEST10 IN FW
033412                          WAIT    2,10                    ;WAIT FOR 10 MS

                        ;
                        ;       ACCESS ALL CSR EXCEPT BSELO/2
                        ;
033422  012702  000001          MOV     #1,R2                   ;START WITH BSEL1
033426  060502                  ADD     R5,R2                   ;GET CORRESPONDING CSR ADDRESS
033430  152722  000210  1$:     BISB    #210,(R2)+              ;ACCESS THIS CSR AND POINTS TO NEXT ONE
033434                          WAIT    2,1                     ;WAIT 1 MS
033444  016504  000000          MOV     SELO(R5),R4             ;GET SELO
033450  042704  000100          BIC     #BIT6,R4                ;CLEAR BIT 6, RESULT IS 0 IF SELO WAS
                                                                ;0 OR 100
033454  001432                  BEQ     3$                      ;RESULT IS NULL, SOMETHING HAPPENS
033456  022702  177002          CMP     #177002,R2              ;NEXT TO TEST IS BSEL2 ?
033462  001001                  BNE     2$                      ;NO, OK CONTINUE
033464  105722                  TSTB    (R2)+                   ;SKIP BSEL2
033466  020227  177037  2$:     CMP     R2,#177037              ;END REACHED ?
033472  101756                  BLOS    1$                      ;NO CONTINUE

                        ;
                        ;       NOW, TEST BSELO/2 ACCESS
```

```
3187                                    ;
3188
3189 033474  060501                         ADD     R5,R1                   ;GET SELECTED CSR ADDRESS
3190 033476  152711  000210                 BISB    #210,(R1)               ;ACCESS THIS CSR
3191 033502                                 WAIT    1,1                     ;WAIT 1 MS
3192 033512  000207                         RETURN                          ;IF TEST OKE
3193 033514  000427                         BR      5$                      ;IF ILLEGAL VECTOR
3194
3195 033516  016503  000004                 MOV     SEL4(R5),R3             ;OTHER,CHECK SEL4=2
3196 033522  032703  000002                 BIT     #BIT1,R3                ;IS IT?
3197 033526  001415                         BEQ     4$                      ;IF NOT
3198
3199 033530                                 ERRHRD  22,EM0021,PGCSR         ;NO INTERRUPT OCCURS
3200 033540  000207                         RETURN
3201
3202 033542                             3$:
3203 033542  160502                         SUB     R5,R2                   ;KEEP ONLY CURRENT OFFSET IN CSR
3204 033544  010237  002404                 MOV     R2,NUM1                 ;SET BAD CSR
3205 033550                                 ERRHRD  23,EM0022,PRCSR         ;ACCESS ON CSR X CAUSES AN INTERRUPT
3206 033560  000207                         RETURN
3207
3208 033562                             4$:
3209 033562                                 ERRHRD  11,EM0011               ;NO KMV11 RESPONSE
3210 033572  000207                         RETURN
3211
3212 033574                             5$:
3213 033574                                 ERRHRD  24,EM0023,PGCSR         ;INTERRUPT AT ILLEGAL VECTOR
3214 033604  000207                         RETURN
3215
3216 033606                             ENDTST
3217                                    .EVEN
3218
```

```
3220 033610                     BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST16 **
                                ;
                                ;
                                ;
3221                            ; TEST OF THE REAL TIME CLOCK
3222                            ;
3223                            ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING THE REAL TIME
3224                            ;               CLOCK BY ITSELF. THE DCT11 TEST RT CLOCK INTERRUPT
3225                            ;               FOR TWO PERIODS OF CLOCK THEN IT DISABLES INTERRUPT
3226                            ;               PERIODICALLY THE HOST POLLS SEL0 FOR ANY
3227                            ;               DCT11 RESPONSE:
3228                            ;                       SEL0 = 0        IF TEST OKE
3229                            ;                       SEL0 = 100      IF ANY ERROR
3230                            ;               IN THE LATTER CASE,THE FOLLOWING CSR'S CON-
3231                            ;               TAIN THE ERROR REPORT AS FOLLOW:
3232                            ;
3233                            ;               - DCT11 INTERRUPTED BEFORE TIME ELAPSED
3234                            ;                       SEL2 = 2
3235                            ;                       SEL4 = CLOCK PERIOD
3236                            ;
3237                            ;               - COUNTER DECREMENTED BUT NO INTERRUPT
3238                            ;                       SEL2 = 4
3239                            ;                       SEL4 = CLOCK PERIOD
3240                            ;
3241                            ;               - COUNTER STOPPED
3242                            ;                       SEL2 = 6
3243                            ;                       SEL4 = CLOCK PERIOD
3244                            ;
3245                            ;               - INTERRUPT AT ILLEGAL VECTOR
3246                            ;                       SEL2 = 10
3247                            ;                       SEL4 = CLOCK PERIOD
3248                            ;
3249                            ;               - DCT11 INTERRUPTED WHILE CLOCK DISABLED
3250                            ;                       SEL2 = 12
3251                            ;
3252                            ;               THEN THE HOST REPORTS CORRESPONDING ERROR.
3253                            ;
3254                            ; MAINT MODE:   2
3255                            ;
3256                            ; KMV ROUTINE:  11
3257                            ;
3258                            ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
3259                            ;               ERROR 11        NO KMV11 RESPONSE
3260                            ;               ERROR 25        INTERRRUPT CCCURS TOO EARLY
3261                            ;               ERROR 26        COUNTER DECREMENTED BUT NO
3262                            ;                               INTERRUPT
3263                            ;               ERROR 27        COUNTER STOPPED
3264                            ;               ERROR 30        KMV11 INTERRUPTED AT ILLEGAL VECTOR
3265                            ;               ERROR 31        INTERRUPT WHILE CLOCK DISABLED
3266                            ;--
```

```
3267
3268
3269 033610                          BGNTST
3270 033610  004737  011116                  CALL    CLRKMV          ;CLEAN CSR'S AND GET CSR ADDRESS
3271 033614                                  MODE    2               ;SELECT MAINTENANCE MODE 1
3272 033620  103002                          BCC     1000$
3273 033622                                  ESCAPE  TST
3274 033626                          1000$:
3275 033626                                  TESTNB  11              ;SELECT TEST11 IN FW
3276 033634                                  WAIT    1,400.          ;WAIT 400MS AND CHECK SELO
3277 033644  000462                          BR      20$             ;SELO IS 0, CORRECT
3278 033646  000406                          BR      1$              ;SELO IS 100, ERROR
3279
3280 033650                                  ERRHRD  11,EM0011       ;IF NO ANSWER
3281 033660                                  ESCAPE  TST
3282
3283 033664  016537  000004  002402  1$:     MOV     SEL4(R5),NUM0   ;GET PARAMETER FOR ERROR MESSAGE
3284 033672                                  DSPACH  5
3285 033700  000406                          BR      2$              ;FOR ERROR 25
3286 033702  000413                          BR      3$              ;FOR ERROR 26
3287 033704  000420                          BR      4$              ;FOR ERROR 27
3288 033706  000425                          BR      5$              ;FOR ERROR 30
3289 033710  000432                          BR      6$              ;FOR ERROR 31
3290
3291 033712                                  ESCAPE  TST
3292
3293 033716                          2$:     ERRHRD  25,EM0024,PRPER ;REPORT
3294 033726                                  ESCAPE  TST
3295
3296 033732                          3$:     ERRHRD  26,EM0025,PRPER ;REPORT
3297 033742                                  ESCAPE  TST
3298
3299 033746                          4$:     ERRHRD  27,EM0026,PRPER ;REPORT
3300 033756                                  ESCAPE  TST
3301
3302 033762                          5$:     ERRHRD  30,EM0027,PRPER ;REPORT
3303 033772                                  ESCAPE  TST
3304
3305 033776                          6$:     ERRHRD  31,EM0030,PRPER ;REPORT
3306 034006                                  ESCAPE  TST
3307
3308 034012  000240                  20$:    NOP
3309 034014                          ENDTST
3310                                 .EVEN
3311
```

```
3313 034016                     BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST17 **
                                ;
                                ;
                                ;
3314                            ; TEST OF TRANSMIT DATA IN INTERRUPT MODE
3315                            ;
3316                            ; DESCRIPTION:  THE HOST ASKS DCT11 FOR TESTING DATA TRANS-
3317                            ;               MISSION IN INTERRUPT MODE.
3318                            ;               THE DCT11 TESTS ALL XMT RELATIVE
3319                            ;               CONTROL BITS AND INTERRUPTS WHILE THE RE-
3320                            ;               CEIVER IS DISABLED. IT TESTS LINE CLOCK COUN-
3321                            ;               TER TOO. THE LINE CLOCK RATIO DIVIDER IS PRE
3322                            ;               VIOUSLY LOADED IN SEL2 BY THE HOST:
3323                            ;                       SEL2 = 226      FOR     72K
3324                            ;                            = 410      FOR     64K
3325                            ;                            = 444      FOR     56K
3326                            ;                            = 504      FOR     48K
3327                            ;                            = 1540     FOR     19.2K
3328                            ;                            = 3440     FOR     9.6K
3329                            ;                            = 24200    FOR     2.4K
3330                            ;               THE HOST SELECTS SPEED BY MEANS OF P.TABLE.
3331                            ;               IN THE CASE OF ALL SPEED REQUESTED,THE TEST
3332                            ;               IS REPEATED FOR EACH SPEED.
3333                            ;               PERIODICALLY THE HOST POLLS SEL0 FOR ANY DCT11 RESPONSE:
3334                            ;                       SEL0 = 0        IF TEST OKE
3335                            ;                       SEL0 = 100      IF ANY ERROR
3336                            ;               IN THE LATTER CASE,THE FOLLOWING CSR'S CON-
3337                            ;               TAIN ERROR REPORTS, SUCH AS:
3338                            ;                       SEL2 = 2  ERROR 32
3339                            ;                            = 4  ERROR 33
3340                            ;                            = 6  ERROR 34
3341                            ;                            = 10 ERROR 40
3342                            ;                            = 12 ERROR 41
3343                            ;                            = 14 ERROR 42
3344                            ;                            = 16 ERROR 43
3345                            ;                       SEL4 = INTERRUPT COUNT
3346                            ;               THEN, THE HOST REPORTS ERROR AND EXIT.
3347                            ;
3348                            ; MAINT MODE:   2
3349                            ;
3350                            ; KMV ROUTINE:  12
3351                            ;
3352                            ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
3353                            ;               ERROR 11        NO KMV11 RESPONSE
3354                            ;               ERROR 32        LINE CLOCK COUNTER STOPPED
3355                            ;               ERROR 33        TIMEOUT WITHOUT INTERRUPT
3356                            ;               ERROR 34        INTERRUPT AT ILLEGAL VECTOR
3357                            ;               ERROR 40        UNEXPECTED INTERRUPT WHILE
3358                            ;                               XMT INTERRUPT DISABLE
3359                            ;               ERROR 41        NO INTERRUPT WHILE WAITING UNDERRUN
```

```
                                        ;               ERROR 42        UNDERRUN INTERRUPT AT ILLEGAL VECTOR
                                        ;               ERROR 43        UNDERRUN INTERRUPT WHILE NOT EXPECTED
                                        ;--
                                        BGNTST
                                        ;       INIT LINE RATE WITH P.TABLE
034016  013701  002362                  MOV     RATE,R1                 ;SAVE CURRENT RATE
034022  022701  000007                  CMP     #7,R1                   ;TEST FOR ALL LINE RATES?
034026  001001                          BNE     1$                      ;IF NOT
034030  005001                          CLR     R1                      ;IF YES, SET FIRST RATE
034032  006301                  1$:     ASL     R1                      ;TO MAKE OFFSET
                                        ;       START TEST
                                2$:
034034  016137  007356  002426          MOV     RATIOT(R1),RATIO        ;READ CURRENT RATE
034042                                  SAVE    1                       ;SAVE R1
034046  004737  011116                  CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS
034052                                  MODE    2                       ;SELECT MAINTENANCE MODE 1
034056  103002                          BCC     1000$
034060                                  ESCAPE  TST
034064                          1000$:
034064  013765  002426  000002          MOV     RATIO,SEL2(R5)          ;INFORM DCT11 FOR RATE
034072                                  TESTNB  12                      ;SELECT TEST12 IN FW
034100                                  GET     1                       ;RESTORE R1
034104                                  WAIT    1,1000.                 ;WAIT 1S AND CHECK SEL0
034114  000473                          BR      20$                     ;SEL0 IS 0, CORRECT
034116  000406                          BR      3$                      ;SEL0 IS 100, ERROR

034120                                  ERRHRD  11,EM0011               ;IF NO RESPONSE
034130                                  ESCAPE  TST

034134                          3$:     DSPACH  10.
034142  000406                          BR      4$                      ;ERROR WHEN LOADING THE LINE COUNTER
034144  000413                          BR      5$                      ;TIMEOUT WITHOUT INTERRUPT
034146  000420                          BR      6$                      ;TRANSMIT INTERRUPT AT ILLEGAL VECTOR
034150  000425                          BR      10$                     ;INTERRUPT OCCURS WHILE DISABLED
034152  000432                          BR      11$                     ;NO INTERRUPT WHILE WAITING UNDERRUN ONE
034154  000437                          BR      12$                     ;UNDERRUN INTERRUPT AT ILLEGAL VECTOR
034156  000444                          BR      13$                     ;UNDERRUN INTERRUPT WHILE NONE EXPECTED

034160                          4$:     ERRHRD  32,EM0031,PRRATE        ;REPORT
034170                                  ESCAPE  TST

034174                          5$:     ERRHRD  33,EM0032,PRITRX        ;REPORT
034204                                  ESCAPE  TST

034210                          6$:     ERRHRD  34,EM0033,PRRATE        ;REPORT
034220                                  ESCAPE  TST

034224                          10$:    ERRHRD  40,EM0037,PRRATE        ;REPORT
034234                                  ESCAPE  TST

034240                          11$:    ERRHRD  41,EM0040,PRRATE        ;REPORT
```

```
3417 034250                                ESCAPE  TST
3418
3419 034254                          12$:  ERRHRD  42,EM0041,PRRATE        ;REPORT
3420 034264                                ESCAPE  TST
3421
3422 034270                          13$:  ERRHRD  43,EM0054,PRRATE        ;REPORT
3423 034300                                ESCAPE  TST
3424
3425                                 ;     NEXT LINE RATE IF REQUESTED
3426
3427 034304                          20$:
3428 034304  022737  000007  002362        CMP     #7,RATE               ;IS IT?
3429 034312  001004                        BNE     21$                   ;IF NOT
3430 034314  005721                        TST     (R1)+                 ;IF YES, SET RATE
3431 034316  022701  000016                CMP     #16,R1                ;ANY-MORE?
3432 034322  001244                        BNE     2$                    ;IF YES, LOOP
3433 034324                          21$:
3434 034324  000240                        NOP
3435 034326                          ENDTST
3436                                 .EVEN
3437
```

```
3439 034330                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST18 **
                                        ;
                                        ;
                                        ;
3440                                    ; TEST OF DATA RECEPTION IN INTERRUPT MODE
3441                                    ;
3442                                    ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING DATA RE-
3443                                    ;               CEPTION BY ITSELF.IN ORDER TO DO THAT, IT
3444                                    ;               PASSES TO THE DCT11 :
3445                                    ;                       SEL2 = LINE CLOCK RATE (AS DEFINED IN
3446                                    ;                              TEST16)
3447                                    ;               THE TEST IS COMPOSED OF 5 SUBTESTS:
3448                                    ;
3449                                    ;               - TEST 18
3450                                    ;               TEST OF INTERRUPTS ON NON-ZERO STATUS AND
3451                                    ;               DATA RECEPTION FOR ONE FRAME SENT,SUCH AS:
3452                                    ;
3453                                    ;                       ---------
3454                                    ;                      )TSOM    )
3455                                    ;                      )1       )
3456                                    ;                      )2       )   >---
3457                                    ;                      )3       )       )
3458                                    ;                      )4       )       )
3459                                    ;                      )5       )       )
3460                                    ;                      )6       )       )
3461                                    ;                      )1       )       )
3462                                    ;                      )2       )       )
3463                                    ;                      )TEOM    )       )
3464                                    ;                      ---------       )
3465                                    ;                      XMT FIFO        )
3466                                    ;                                      )
3467                                    ;                      ---------       )
3468                                    ;                      )RSOM!1  )      )
3469                                    ;                      )2       )   <---
3470                                    ;                      )3       )
3471                                    ;                      )4       )
3472                                    ;                      )5       )
3473                                    ;                      )6       )
3474                                    ;                      )1       )
3475                                    ;                      )REOM!2  )
3476                                    ;                      ---------
3477                                    ;                      RCV FIFO
3478                                    ;
3479                                    ;
3480                                    ;
3481                                    ;               - TEST 19
3482                                    ;               TEST OF TWO FRAMES CONSECUTIVELY SENT BUT
3483                                    ;               ONLY THE SECOND ONE IS TAKEN INTO ACCOUNT.
3484                                    ;
```

```
                        ;       ---------
                        ;       )TSOM     )
                        ;       )1        )
                        ;       )2        )
                        ;       )3        )
                        ;       )4        )
                        ;       )5        )
                        ;       )TEOM     )   >--------------->          ---------
                        ;       )TSOM     )                              )RSOM!1   )
                        ;       )1        )                              )REOM!5   )
                        ;       )2        )                              )RSOM!1   )
                        ;       )3        )                              )2        )
                        ;       )4        )                              )3        )
                        ;       )5        )                              )4        )
                        ;       )6        )                              )5        )
                        ;       )1        )                              )6        )
                        ;       )2        )                              )1        )
                        ;       )TEOM     )                              )REOM!2   )
                        ;       ---------                                ---------
                        ;       XMT FIFO                                 RCV FIFO
                        ;
                        ;
                        ;               - TEST 20
                        ;               AN ABORT CHARACTER IS RECEIVED DUE TO A TRANSMIT
                        ;               UNDERRUN AND ANOTHER ONE DUE TO A REAL ONE TRANSMITTED.
                        ;
                        ;       ---------
                        ;       )TSOM     )                              ---------
                        ;       )1        )       ------->               )RSOM!1   )
                        ;       )2        )                              )2        )
                        ;       )3        )                              )3        )
                        ;       )4        )                              )4        )
                        ;       )5        )
                        ;       )6        )
                        ;       )1        )
                        ;       THEN STOP
                        ;       UNTIL UNDERRUN  ------->                 RABO+REOM!4
                        ;
                        ;       )TSOM     )
                        ;       )1        )       ------->
                        ;       )2        )                              RSOM!1
                        ;       )3        )                              RABO+REOM!1
                        ;       )4        )
                        ;       )TXAB     )
                        ;       ---------
                        ;       XMT FIFO
                        ;
```

```
3535                    ;
3536                    ;               - TEST 21
3537                    ;               TEST OF NON ZERO STATUS INTERRUPT IN CASE OF
3538                    ;               CRC ERROR. THE COM DEVICE
3539                    ;               IS INITIALIZED TO SEND A FLAG INSTEAD OF AN
3540                    ;               ABORT(IDLE=1) ON TXAB SET
3541                    ;
3542                    ;       ---------
3543                    ;       )TSOM   )                                       ----------------
3544                    ;       )1      )                                       )RSOM!1         )
3545                    ;       )2      )                                       )2              )
3546                    ;       )3      )                                       )3              )
3547                    ;       )4      ) --> FLAG -->                          )ERRCHK!REOM 4  )
3548                    ;       )5      )                                       ----------------
3549                    ;       )6      )
3550                    ;       )TABO   )
3551                    ;       ---------                                       RCV FIFO
3552                    ;       XMT FIFO
3553                    ;
3554                    ;
3555                    ;
3556                    ;               - TEST 22
3557                    ;               TEST OF RECEIVE OVERRUN
3558                    ;
3559                    ;       ---------                               ---------
3560                    ;       )TSOM   )  >--------------->   RSOM )1      )
3561                    ;       )1      )                      ROR  )2      )
3562                    ;       )2      )                           )3      )
3563                    ;       )3      )                           )4      )
3564                    ;
3565                    ;       UP TO 68 INCREMENTAL DATA       UP TO 64 DATA IN
3566                    ;
3567                    ;       )104    )                       )       )
3568                    ;       )TEOM   )                       ---------
3569                    ;       ---------
3570                    ;
3571                    ;
```

```
                        ;               PERIODICALLY THE HOST POLLS SEL0 FOR ANY DCT11
                        ;               RESPONSE:
                        ;                       SEL0 = 0        FOR TEST OKE
                        ;                       SEL0 = 100      IF ANY ERROR
                        ;               IN THE LATTER CASE SEL2 POINTS TO THE ERROR,
                        ;               POSSIBLE ERRORS ARE :
                        ;                       SEL2 = 2        FOR ERROR 32
                        ;                            = 4        FOR ERROR 45
                        ;                            = 6        FOR ERROR 40
                        ;                            = 10       FOR ERROR 46
                        ;                            = 12       FOR ERROR 47
                        ;                            = 14       FOR ERROR 50
                        ;                            = 16       FOR ERROR 51
                        ;                            = 20       FOR ERROR 35
                        ;                            = 22       FOR ERROR 43
                        ;                       SEL4 = OBTAINED STATUS&DATA
                        ;                       SEL6 = EXPECTED ONE
                        ;                       SEL10= EVENT FLAGS
                        ;
                        ; MAINT MODE:  2
                        ;
                        ; KMV ROUTINE: 13              FOR TEST 18
                        ;             14              FOR TEST 19
                        ;             15              FOR TEST 20
                        ;             16              FOR TEST 21
                        ;             17              FOR TEST 22
                        ;
                        ; REPORTS:     ERROR 2         MASTER CLEAR FAILS TO RESET
                        ;             ERROR 11        NO KMV11 RESPONSE
                        ;             ERROR 32        LINE CLOCK FAILURE
                        ;             ERROR 40        TRANSMIT INTERRUPT WHILE DISABLED
                        ;             ERROR 35        UNDERRUN INTERRUPT
                        ;             ERROR 43        UNEXPECTED UNDERRUN INTERRUPT AFTER ACK
                        ;             ERROR 45        XMT/RCV PROCESSING FAILS
                        ;             ERROR 46        UNEXPECTED STATUS RECEIVED
                        ;             ERROR 47        UNEXPECTED DATA RECEIVED
                        ;             ERROR 50        RECEIVE INTERRUPT WHILE DISABLED
                        ;             ERROR 51        INTERRUPT AT ILLEGAL VECTOR
                        ;
                        ;--

                        ;       TEST 18
034330                  BGNTST
034330  004737  034626          1$:     CALL    STRSB0          ;CALL START SUBROUTINE 0
034334  004737  034646                  CALL    STRSB1          ;CALL START SUBROUTINE 1
034340                          1$:     TESTNB  13              ;SELECT TEST13 IN FW
034346  004737  034710                  CALL    T18B00          ;CHECK ERROR
034352  022737  000007  002362          CMP     #7,RATE         ;NEXT LINE RATE REQUESTED ?
034360  001004                          BNE     2$              ;IF NOT
034362  005721                          TST     (R1)+           ;IF YES, SET RATE
034364  022701  000016                  CMP     #16,R1          ;ANY-MORE?
034370  001361                          BNE     1$              ;IF YES, LOOP
034372                          2$:
034372  000240                          NOP
034374                  ENDTST
```

```
                                        BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                          ** TEST19 **
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;--
                                        BGNTST
                                                CALL    STRSB0                  ;CALL START SUBROUTINE 0
                                        1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
                                                TESTNB  14                      ;SELECT TEST14 IN FW
                                                CALL    T18B00                  ;CHECK ERROR
                                                CMP     #7,RATE                 ;NEXT LINE RATE REQUESTED ?
                                                BNE     2$                      ;IF NOT
                                                TST     (R1)+                   ;IF YES, SET RATE
                                                CMP     #16,R1                  ;ANY-MORE?
                                                BNE     1$                      ;IF YES, LOOP
                                        2$:
                                                NOP
                                        ENDTST

                                        BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                          ** TEST20 **
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;--
                                        BGNTST
                                                CALL    STRSB0                  ;CALL START SUBROUTINE 0
                                        1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
                                                TESTNB  15                      ;SELECT TEST15 IN FW
                                                CALL    T18B00                  ;CHECK ERROR
                                                CMP     #7,RATE                 ;NEXT LINE RATE REQUESTED ?
                                                BNE     2$                      ;IF NOT
                                                TST     (R1)+                   ;IF YES, SET RATE
                                                CMP     #16,R1                  ;ANY-MORE?
                                                BNE     1$                      ;IF YES, LOOP
                                        2$:
                                                NOP
                                        ENDTST

                                        BADHEAD
```

```
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST21 **
                                        ;
                                        ;
                                        ;
                                        ;
3667                                    ;
3668                                    ;--
3669
3670 034512                             BGNTST
3671 034512  004737  034626                    CALL    STRSB0          ;CALL START SUBROUTINE 0
3672 034516  004737  034646             1$:    CALL    STRSB1          ;CALL START SUBROUTINE 1
3673 034522                                    TESTNB  16              ;SELECT TEST16 IN FW
3674 034530  004737  034710                    CALL    T18B00          ;CHECK ERROR
3675 034534  022737  000007  002362            CMP     #7,RATE         ;NEXT LINE RATE REQUESTED ?
3676 034542  001004                            BNE     2$              ;IF NOT
3677 034544  005721                            TST     (R1)+           ;IF YES, SET RATE
3678 034546  022701  000016                    CMP     #16,R1          ;ANY-MORE?
3679 034552  001361                            BNE     1$              ;IF YES, LOOP
3680 034554                             2$:
3681 034554  000240                            NOP
3682 034556                             ENDTST
3683
3684 034560                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST22 **
                                        ;
                                        ;
                                        ;
                                        ;
3685                                    ;
3686                                    ;--
3687
3688 034560                             BGNTST
3689 034560  004737  034626                    CALL    STRSB0          ;CALL START SUBROUTINE 0
3690 034564  004737  034646             1$:    CALL    STRSB1          ;CALL START SUBROUTINE 1
3691 034570                                    TESTNB  17              ;SELECT TEST17 IN FW
3692 034576  004737  034710                    CALL    T18B00          ;CHECK ERROR
3693 034602  022737  000007  002362            CMP     #7,RATE         ;NEXT LINE RATE REQUESTED ?
3694 034610  001004                            BNE     2$              ;IF NOT
3695 034612  005721                            TST     (R1)+           ;IF YES, SET RATE
3696 034614  022701  000016                    CMP     #16,R1          ;ANY-MORE?
3697 034620  001361                            BNE     1$              ;IF YES, LOOP
3698 034622                             2$:
3699 034622  000240                            NOP
3700 034624                             ENDTST
3701
3702
3703                                    ;
```

```
3704                                    ; THESE SUBROUTINES ARE CALLED AT THE BEGINNING OF EACH TEST.
3705                                    ; THEY INIT THE LINE RATE, INIT THE KMV11-C AND SELECT THE MODE 2.
3706                                    ;
3707 034626                            STRSB0:
3708 034626  013701  002362                    MOV     RATE,R1                 ;SAVE CURRENT RATE
3709 034632  022701  000007                    CMP     #7,R1                   ;TEST FOR ALL LINE RATES?
3710 034636  001001                            BNE     1$                      ;IF NOT
3711 034640  005001                            CLR     R1                      ;IF YES, SET FIRST RATE
3712 034642  006301                    1$:     ASL     R1                      ;TO MAKE OFFSET
3713 034644  000207                            RETURN
3714
3715 034646  016137  007356  002426  STRSB1: MOV     RATIOT(R1),RATIO        ;READ CURRENT RATE
3716 034654                                    SAVE    1                       ;SAVE R1
3717 034660  004737  011116                    CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS
3718 034664                                    MODE    2                       ;SELECT MAINTENANCE MODE 1
3719 034670  103001                            BCC     1000$
3720 034672  000207                            RETURN
3721 034674                            1000$:
3722 034674  013765  002426  000002            MOV     RATIO,SEL2(R5)          ;INFORM DCT11 FOR RATE
3723 034702                                    GET     1                       ;RESTORE  R1
3724 034706  000207                            RETURN
3725                                    ;
3726                                    ;       CHECK ERROR
3727                                    ;
3728 034710                            T18800:
3729 034710                                    WAIT    1,1000.                 ;WAIT 400MS AND CHECK SEL0
3730 034720  000207                            RETURN                          ;SEL0 IS 0, CORRECT
3731 034722  000405                            BR      3$                      ;SEL0 IS 100, ERROR
3732
3733 034724                                    ERRHRD  11,EM0011               ;IF NO RESPONSE
3734 034734  000207                            RETURN
3735
3736 034736                            3$:     DSPACH  9.
3737 034744  000410                            BR      4$                      ;ERROR WHEN LOADING THE LINE COUNTER
3738 034746  000433                            BR      5$                      ;GENERAL ERROR
3739 034750  000413                            BR      6$                      ;XMT INTERRUPT WHILE DISABLED
3740 034752  000436                            BR      7$                      ;UNEXPECTED RECEIVE STATUS INTERRUPT
3741 034754  000450                            BR      8$                      ;UNEXPECTED RECEIVE DATA INTERRUPT
3742 034756  000462                            BR      9$                      ;RECEIVE INTERRUPT WHILE DISABLED
3743 034760  000474                            BR      10$                     ;INTERRUPT AT ILLEGAL VECTOR
3744 034762  000413                            BR      11$                     ;UNDERRUN INTERRUPT
3745 034764  000417                            BR      12$                     ;UNEXPECTED INTERRUPT AFTER ACK
3746
3747 034766                            4$:     ERRHRD  32,EM0031,PRRATE        ;REPORT
3748 034776  000207                            RETURN
3749
3750 035000                            6$:     ERRHRD  40,EM0037,PREVT         ;REPORT
3751 035010  000207                            RETURN
3752
3753 035012                            11$:    ERRHRD  35,EM0034,PREVT         ;REPORT
3754 035022  000207                            RETURN
3755
3756 035024                            12$:    ERRHRD  43,EM0042,PREVT         ;REPORT
3757 035034  000207                            RETURN
3758
3759 035036                            5$:     ERRHRD  45,EM0044,PREVT         ;REPORT
3760 035046  000207                            RETURN
```

```
3761
3762 035050  016537  000004  002412  7$:     MOV     SEL4(R5),BDDAT          ;GET RCV STATUS AND DATA REGISTER
3763 035056  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;GET EXPECTED ONE
3764 035064                                  ERRHRD  46,EM0045,PRDAT         ;REPORT
3765 035074  000207                          RETURN
3766
3767 035076  016537  000004  002412  8$:     MOV     SEL4(R5),BDDAT          ;GET RCV STATUS AND DATA REGISTER
3768 035104  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;GET EXPECTED ONE
3769 035112                                  ERRHRD  47,EM0046,PRDAT         ;REPORT
3770 035122  000207                          RETURN
3771
3772 035124  016537  000004  002412  9$:     MOV     SEL4(R5),BDDAT          ;GET RCV STATUS AND DATA REGISTER
3773 035132  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;GET EXPECTED ONE
3774 035140                                  ERRHRD  50,EM0047,PRDAT         ;REPORT
3775 035150  000207                          RETURN
3776
3777 035152                          10$:    ERRHRD  51,EM0050,PREVT         ;REPORT
3778 035162  000207                          RETURN
3779
3780                                  .EVEN
3781
```

```
   3783 035164                          BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST23 **
                                        ;
                                        ;
                                        ;
                                        ;
   3784                                 ; TEST OF DMA IN (TRX) TRANSFER USING A SHORT FRAME OF DATA
   3785                                 ;
   3786                                 ; DESCRIPTION:  THE HOST LOADS A FRAME OF 36 PATTERNS INSIDE
   3787                                 ;               ITS MEMORY. THEN IT ASKS THE DCT11  TO TRANS
   3788                                 ;               FER BY DMA THE FRAME INTO THE XMT  FIFO  AND
   3789                                 ;               TO TRANSMIT, RECEIVE BACK THIS  FRAME. DCT11
   3790                                 ;               CHECKS THE RECEIVED FRAME. IN ORDER TO  INIT
   3791                                 ;               DMA TRANSFER THE  HOST PASSES PARAMETERS  TO
   3792                                 ;               THE DCT11, SUCH AS:
   3793                                 ;                       SEL2 = LINE CLOCK RATE
   3794                                 ;                       SEL4 = DMA IN BYTE COUNT(2'S COMP)
   3795                                 ;                       SEL6 = DMA IN BUS ADDRESS(EVEN)
   3796                                 ;                       SEL10= EXT ADDRESS
   3797                                 ;                       SEL12= SUBTEST CODE
   3798                                 ;               THE FRAME TO TRANSFER IS:
   3799                                 ;                       ALL 0
   3800                                 ;                       ALL 1
   3801                                 ;                       052525
   3802                                 ;                       125252
   3803                                 ;                       ROTATING 0
   3804                                 ;                       ROTATING 1
   3805                                 ;               NOTICE  THAT  DMA IN  IS WORD WIDE BUT, EACH
   3806                                 ;               WORD IS LOADED IN FIFO BYTE BY BYTE.
   3807                                 ;               PERIODICALLY THE HOST POLLS SELO FOR ANY RES
   3808                                 ;               PONSE FROM THE KMV11:
   3809                                 ;                       SELO = 0 IF TEST CORRECTLY ENDED
   3810                                 ;                       SELO = 100 IF ANY ERROR
   3811                                 ;               IN THE LATTER CASE:
   3812                                 ;                       SEL2 =  2       FOR ERROR 32
   3813                                 ;                               4       FOR ERROR 45
   3814                                 ;                               6       FOR ERROR 54
   3815                                 ;                               10      FOR ERROR 55
   3816                                 ;                               12      FOR ERROR 56
   3817                                 ;                               14      FOR ERROR 46
   3818                                 ;                               16      FOR ERROR 57
   3819                                 ;                               20      FOR ERROR 50
   3820                                 ;                               22      FOR ERROR 51
   3821                                 ;                               24      FOR ERROR 35
   3822                                 ;                       SEL10=  FLAGWORD DESCRIBING PROCESSING
   3823                                 ;                       SEL12=  CURRENT TRX DMA BYTE COUNT
   3824                                 ;                       SEL14 =  CURRENT TRX DMA BUFFER ADDRESS (LOW)
   3825                                 ;                       SEL16 =  CURRENT TRX DMA BUFFER ADDRESS (HIGH)
   3826                                 ;                     IF ERROR 46,50,57 :
   3827                                 ;                       SEL4=  RCV STATUS AND DATA REGISTER
   3828                                 ;                       SEL6=  EXPECTED VALUE
   3829                                 ;
```

```
                        ;                THEN THE HOST BUILDS  ERROR REPORT  AND EXIT
                        ;
                        ; MAINT MODE:    2
                        ;
                        ; KMV ROUTINE:   20
                        ;
                        ; REPORTS:       ERROR 2         MASTER CLEAR FAILS TO RESET
                        ;                ERROR 11        NO KMV11 RESPONSE
                        ;                ERROR 32        LINE COUNTER ERROR
                        ;                ERROR 35        UNDERRUN INTERRUPT OCCURS
                        ;                ERROR 45        TRX/RCV ERROR
                        ;                ERROR 46        UNEXPECTED STATUS RECEIVED
                        ;                ERROR 50        UNEXPECTED RCV INT. WHILE DISABLED
                        ;                ERROR 51        INTERRUPT AT ILLEGAL VECTOR
                        ;                ERROR 54        UNEXPECTED END OF RCV DMA INT.
                        ;                ERROR 55        END OF TRX DMA WITHOUT BIT "BCNTIN" SET
                        ;                ERROR 56        TIMEOUT DURING TRX DMA
                        ;                ERROR 57        DATA COMPARE ERROR DURING TRX DMA
                        ;--

035164                  BGNTS:
035164  004737  034626          CALL    STRSB0                  ;CALL START SUBROUTINE 0
035170  004737  034646  1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
035174  012765  007204  000006  MOV     #PATTRN,SEL6(R5)        ;LOAD TRX BUFFER ADDRESS
035202  005065  000010          CLR     SEL10(R5)               ;LOAD EXT. ADDRESS
035206  012765  000112  000004  MOV     #PATLGH,SEL4(R5)        ;LOAD BYTE COUNT
035214  005465  000004          NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT
035220  005065  000012          CLR     SEL12(R5)               ;LOAD SUBTEST CODE FOR FW TEST
035224  005037  002364          CLR     NXMTST                  ;CLEAN NXM TEST FLAG
035230  004737  035470          CALL    DMAINS                  ;CALL COMMON SUBROUTINE
035234  103410                  BCS     20$                     ;ERROR
035236  022737  000007  002362  CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
035244  001004                  BNE     20$                     ;NO, QUIT THIS IS THE END
035246  005721                  TST     (R1)+                   ;NEXT SPEED
035250  022701  000016          CMP     #16,R1                  ;END OF TABLE ?
035254  001345                  BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
035256  000240          20$:    NOP
035260                          ENDTST
035262                  BADHEAD
                        ;**
                        ;
                        ;
                        ;
                        ;                              ** TEST24 **
                        ;
                        ;
                        ;
                        ;
                        ; TEST OF DMA TRANSFER IN (TRX) USING A LONG FRAME OF DATA
                        ;
                        ; A 1020. BYTE LONG FRAME IS TRANSMITTED IN DMA MODE WHILE THE RECEPTION
                        ; IS DONE IN INTERRUPT MODE.
                        ;
                        ; INPUT: SEL2 = LINE CLOCK RATE
```

```
                        ;       SEL4 = DMA IN BYTE COUNT (2'S COMP)
                        ;       SEL6 = DMA IN BUFFER ADDRESS
                        ;       SEL10= EXT ADDRESS
                        ;       SEL12= SUBTEST CODE (*2)
                        ;--
035262                  BGNTST
035262  004737  034626          CALL    STRSB0                  ;CALL START SUBROUTINE 0
035266  004737  034646  1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
035272  012765  002632  000006  MOV     #TRXBUF,SEL6(R5)        ;LOAD TRX BUFFER ADDRESS
035300  005065  000010          CLR     SEL10(R5)               ;LOAD EXT. ADDRESS
035304  012765  001774  000004  MOV     #TDMALG,SEL4(R5)        ;LOAD BYTE COUNT
035312  005465  000004          NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT
035316  012765  000002  000012  MOV     #2,SEL12(R5)            ;LOAD SUBTEST CODE FOR FW TEST
035324  005037  002364          CLR     NXMTST                  ;CLEAN NXM TEST FLAG
035330  004737  035470          CALL    DMAINS                  ;CALL COMMON SUBROUTINE
035334  103410                  BCS     20$                     ;ERROR
035336  022737  000007  002362  CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
035344  001004                  BNE     20$                     ;NO, QUIT THIS IS THE END
035346  005721                  TST     (R1)+                   ;NEXT SPEED
035350  022701  000016          CMP     #16,R1                  ;END OF TABLE ?
035354  001344                  BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
035356  000240          20$:    NOP
035360                  ENDTST

035362                  BADHEAD
                        ;++
                        ;
                        ;
                        ;
                        ;                       ** TEST25 **
                        ;
                        ;
                        ;
                        ;       >
                        ;
                        ; NON EXISTENT MEMORY TEST DURING TRANSMIT DMA TRANSFER
                        ;
                        ;--
035362                  BGNTST
035362  012737  000001  002364          MOV     #1,NXMTST               ;SET FLAG TO MARK NXM TEST
035370  004737  041542                  JSR     PC,$SIZE                ;GET A REAL NON EXISTENT MEMORY ADDRESS
035374  004737  034626                  CALL    STRSB0                  ;START SUBROUTINE 0
035400  004737  034646          1$:     CALL    STRSB1                  ;START SUBROUTINE 1
035404  013765  002526  000006          MOV     NXMLOW,SEL6(R5)         ;LOAD LOW NXM ADDRESS /V01.04/
035412  013765  002530  000010          MOV     NXMHIG,SEL10(R5)        ;LOAD HIGH NXM ADDRESS /V01.04/
035420  012765  000012  000004          MOV     #10.,SEL4(R5)           ;LOAD BYTE COUNT
035426  005465  000004                  NEG     SEL4(R5)                ;GET IT IN 2'S COMPL. FORM
035432  005065  000012                  CLR     SEL12(R5)               ;SUBTEST 1 SELECTED IN FW
035436  004737  035470                  CALL    DMAINS                  ;CALL COMMON SUBROUTINE
035442  103410                          BCS     20$                     ;UNEXPECTED RESULT
035444  022737  000007  002362          CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
035452  001004                          BNE     20$                     ;NO, QUIT THIS IS THE END
035454  005721                          TST     (R1)+                   ;NEXT SPEED
035456  022701  000016                  CMP     #16,R1                  ;END OF TABLE ?
035462  001346                          BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
035464  000240                  20$:    NOP
```

```
3924 035466                                ENDTST
3925
3926 035470                          DMAINS: TESTNB  20                  ;LOAD KMV ROUTINE NUMBER
3927
3928 035476                                  WAIT    1,1000.             ;WAIT
3929 035506  000542                         BR      25$                 ;SEL0 = 0, CORRECT ANSWER
3930 035510  000406                         BR      10$                 ;SEL0 = 100, ERROR
3931
3932 035512                                  ERRHRD  11,EM0011           ;IF NO RESPONSE
3933 035522  000261                    9$:   SEC
3934 035524  000207                         RETURN
3935
3936 035526  005737  002364          10$:   TST     NXMTST              ; LOOK FOR NXM TEST
3937 035532  001404                         BEQ     26$                 ; NO DISPATCH
3938 035534  122765  000012  000002        CMPB    #12,SEL2(R5)        ; REAL TIMOUT ?
3939 035542  001124                         BNE     25$
3940
3941 035544                          26$:   DSPACH  10.
3942 035552  000411                         BR      11$                 ;LINE COUNTER ERROR
3943 035554  000415                         BR      12$                 ;TRX/RCV ERROR
3944 035556  000421                         BR      13$                 ;UNEXPECTED END OF RCV DMA INT.
3945 035560  000425                         BR      14$                 ;END OF TRX DMA WITHOUT BIT "BCNTIN" SET
3946 035562  000431                         BR      15$                 ;TIMEOUT DURING TRX DMA
3947 035564  000440                         BR      16$                 ;UNEXPECTED STATUS RECEIVED
3948 035566  000452                         BR      17$                 ;DATA COMPARE ERROR DURING TRX DMA
3949 035570  000464                         BR      18$                 ;UNEXPECTED RCV INT. WHILE DISABLED
3950 035572  000476                         BR      19$                 ;INTERRUPT AT ILLEGAL VECTOR
3951 035574  000502                         BR      20$                 ;UNDERRUN INTERRUPT OCCURS
3952
3953 035576                          11$:   ERRHRD  32,EM0032,PRRATE
3954 035606  000745                         BR      9$
3955
3956 035610                          12$:   ERRHRD  45,EM0044,PDMAT3
3957 035620  000740                         BR      9$
3958
3959 035622                          13$:   ERRHRD  54,EM0054,PDMAT1
3960 035632  000733                         BR      9$
3961
3962 035634                          14$:   ERRHRD  55,EM0055,PDMAT1
3963 035644  000726                         BR      9$
3964
3965 035646  005737  002364          15$:   TST     NXMTST                          ;NXM TEST RUNNING ?
3966 035652  001070                         BNE     5$                              ;YES , SUCCESS
3967 035654                                  ERRHRD  56,EM0056,PDMAT1
3968 035664  000716                         BR      9$
3969
3970 035666  016537  000004  002412  16$:   MOV     SEL4(R5),BDDAT                  ;LOAD RECEIVED STATUS&DATA
3971 035674  016537  000006  002410          MOV     SEL6(R5),GDDAT                  ;LOAD EXPECTED ONE
3972 035702                                  ERRHRD  46,EM0045,PDMAT2
3973 035712  000703                         BR      9$
3974
3975 035714  016537  000004  002412  17$:   MOV     SEL4(R5),BDDAT                  ;LOAD RECEIVED STATUS&DATA
3976 035722  016537  000006  002410          MOV     SEL6(R5),GDDAT                  ;LOAD EXPECTED ONE
3977 035730                                  ERRHRD  57,EM0057,PDMAT2
3978 035740  000670                         BR      9$
3979
3980 035742  016537  000004  002412  18$:   MOV     SEL4(R5),BDDAT                  ;LOAD RECEIVED STATUS&DATA
```

```
3981 035750  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;LOAD EXPECTED ONE
3982 035756                                  ERRHRD  50,EM0047,PDMAT2
3983 035766  000655                          BR      9$
3984
3985 035770                          19$:    ERRHRD  51,EM0050,PDMAT3
3986 036000  000650                          BR      9$
3987
3988 036002                          20$:    ERRHRD  35,EM0034,PDMAT3
3989 036012  000643                          BR      9$
3990
3991 036014  005737  002364          25$:    TST     NXMTST                  ;NXM TEST RUNNING ?
3992 036020  001405                          BEQ     5$                      ;NO, REAL SUCCESS
3993 036022                                  ERRHRD  65,EM0065               ;UNEXPECTED SUCCESS
3994 036032  000633                          BR      9$
3995 036034  000241                  5$:     CLC
3996 036036  000207                          RETURN
```

```
                        BADHEAD
                        ;**
                        ;
                        ;
                        ;
                        ;                       ** TEST26 **
                        ;
                        ;
                        ;
3999                    ; TEST OF DMA OUT (RCV) TRANSFER USING A SHORT FRAME OF DATA
4000                    ;
4001                    ; DESCRIPTION:  THE HOST PREPARES A  SPACE IN ITS MEMORY FOR
4002                    ;               THE TRANSFERRED FRAME. THEN, THE HOST ASKS
4003                    ;               THE DCT11 FOR PROCESSING. IN ORDER TO INIT
4004                    ;               DMA OUT, THE HOST PASSES PARAMETERS TO THE
4005                    ;               KMV, SUCH AS:
4006                    ;                       SEL2 = LINE CLOCK RATE
4007                    ;                       SEL4 = DMA OUT BYTE COUNT(2'S COMP.)
4008                    ;                       SEL6 = RECEIVE BUFFER ADDRESS(EVEN)
4009                    ;                       SEL10= EXT. ADDRESS
4010                    ;                       SEL12= SUBTEST CODE (0 OR 2)
4011                    ;               THE DCT11 SENDS , RECEIVES AND TRANSFERS BY
4012                    ;               DMA THE SAME FRAME AS NOTICE IN TEST 23.
4013                    ;               DMA OUT IS BYTE WIDE.
4014                    ;               PERIODICALLY THE HOST POLLS SEL0 FOR ANY RES
4015                    ;               PONSE FROM THE KMV11:
4016                    ;                       SEL0 = 0 IF TEST OKE
4017                    ;                       SEL0 = 100 IF ANY ERROR
4018                    ;               IN THE FORMER CASE, THE HOST CHECK THE FRAME
4019                    ;               LOADED BY DMA AND REPORT ERROR 70 IF A DATA
4020                    ;               COMPARE ERROR.
4021                    ;               IN THE LATTER CASE,
4022                    ;                       SEL2 =  2       FOR ERROR 32
4023                    ;                               4       FOR ERROR 45
4024                    ;                               6       FOR ERROR 61
4025                    ;                               10      FOR ERROR 62
4026                    ;                               12      FOR ERROR 63
4027                    ;                               14      FOR ERROR 46
4028                    ;                               16      FOR ERROR 51
4029                    ;                               20      FOR ERROR 35
4030                    ;                       SEL4 =  RCV STATUS AND DATA REGISTER
4031                    ;                       SEL6 =  EXPECTED STATUS AND DATA REGISTER
4032                    ;                       SEL10=  FLAGWORD
4033                    ;                       SEL12=  CURRENT RCV BYTE COUNT
4034                    ;                       SEL14 = CURRENT RCV BUFFER ADDRESS (LOW)
4035                    ;                       SEL16 = CURRENT RCV BUFFER ADDRESS (HIGH)
4036                    ;               THEN THE HOST BUILDS ERROR REPORTS DEPENDING
4037                    ;               ON SEL0 THEN EXIT.
4038                    ;
4039                    ; MAINT MODE:   2
4040                    ;
4041                    ; KMV ROUTINE:  21
4042                    ;
4043                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
4044                    ;               ERROR 11        NO KMV11 RESPONSE
```

```
4045                                    ;               ERROR 35        UNDERRUN INTERRUPT OCCURS
4046                                    ;               ERROR 45        TRX/RCV ERROR
4047                                    ;               ERROR 46        UNEXPECTED RECEIVED STATUS
4048                                    ;               ERROR 61        UNEXPECTED END OF TRX DMA
4049                                    ;               ERROR 62        RCV BYTE COUNT OVERFLOW
4050                                    ;               ERROR 63        TIMEOUT DURING RCV DMA
4051                                    ;               ERROR 51        INTERRUPT AT ILLEGAL VECTOR
4052                                    ;--
4053
4054
4055 036040                            BGNTST
4056 036040  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4057 036044  004737  034646            1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
4058 036050                                    SAVE    1
4059 036054  012701  004626                    MOV     #RCVBUF,R1              ;GET RECEIVE BUFFER
4060 036060  012737  000776  002470            MOV     #1020./2,BUFLEN         ;LOAD CORREPONDING LENGTH
4061 036066  004737  011052                    CALL    CLEAR                   ;CLEAR THE RECEIVE BUFFER
4062 036072                                    GET     1
4063 036076  012765  004626  000006            MOV     #RCVBUF,SEL6(R5)        ;LOAD RCV BUFFER ADDRESS
4064 036104  005065  000010                    CLR     SEL10(R5)               ;LOAD EXT. ADDRESS
4065 036110  012765  000112  000004            MOV     #PATLGH,SEL4(R5)        ;LOAD BYTE COUNT
4066 036116  005465  000004                    NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT VALUE
4067 036122  005065  000012                    CLR     SEL12(R5)               ;SELECT SUBTEST 0 IN FW
4068 036126  005037  002364                    CLR     NXMTST                  ;CLEAR NXM FLAG
4069 036132  004737  036610                    CALL    DMAOSB                  ;CALL COMMON SUBROUTINE
4070 036136  103443                            BCS     20$                     ;ERROR
4071 036140                                    SAVE    1
4072 036144  012701  007206                    MOV     #PATTRN+2,R1            ;GET EXPECTED BUFFER ADDRESS
4073 036150  012702  004626                    MOV     #RCVBUF,R2              ;GET RECEIVE BUFFER ADDRESS
4074 036154  012737  000106  002470            MOV     #PATLGH-4,BUFLEN        ;LOAD LENGTH OF BUFFER TO COMPARE
4075 036162  004737  010444                    CALL    COMPRB                  ;COMPARE BOTH BUFFERS BYTE PER BYTE
4076 036166  103015                            BCC     15$                     ;SUCCESS
4077 036170  012737  000020  002470            MOV     #CSRLEN,BUFLEN          ;REINIT BUFLEN
4078 036176  012703  007312                    MOV     #PATTRN+PATLGH-4,R3     ;LOAD TRANSMIT END BUFFER ADDRESS
4079 036202                                    ERRHRD  60,EM0060,PRDMAR        ;DATA COMPARE ERROR DURING RCV DMA
4080 036212                                    GET     1
4081 036216                                    ESCAPE  TST
4082 036222                            15$:    GET     1
4083 036226  022737  000007  002362            CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4084 036234  001004                            BNE     20$                     ;NO, QUIT THIS IS THE END
4085 036236  005721                            TST     (R1)+                   ;NEXT SPEED
4086 036240  022701  000016                    CMP     #16,R1                  ;END OF TABLE ?
4087 036244  001277                            BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4088 036246  012737  000020  002470    20$:    MOV     #CSRLEN,BUFLEN          ;REINIT BUFLEN
4089 036254  000240                            NOP
4090 036256                                    ENDTST
4091 036260                            BADHEAD
                                       ;++
                                       ;
                                       ;
                                       ;
                                       ;                       ++ TEST27 ++
                                       ;
                                       ;
                                       ;
                                       ;
```

```
4092                                    ; TEST OF DMA OUT (RCV) TRANSFER USING A LONG FRAME OF DATA
4093                                    ;---
4094 036260                             BGNTST
4095 036260  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4096 036264  004737  034646            1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
4097 036270                                    SAVE    1
4098 036274  012701  004626                    MOV     @RCVBUF,R1              ;GET RECEIVE BUFFER
4099 036300  012737  000776  002470            MOV     @1020./2,BUFLEN         ;LOAD CORREPONDING LENGTH
4100 036306  004737  011052                    CALL    CLEAR                   ;CLEAR THE RECEIVE BUFFER
4101 036312                                    GET     1
4102 036316  012765  004626  000006            MOV     @RCVBUF,SEL6(R5)        ;LOAD RCV BUFFER ADDRESS
4103 036324  005065  000010                    CLR     SEL10(R5)               ;LOAD EXT. ADDRESS
4104 036330  012765  001774  000004            MOV     @1020.,SEL4(R5)         ;LOAD BYTE COUNT
4105 036336  005465  000004                    NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT VALUE
4106 036342  012765  000002  000012            MOV     @2,SEL12(R5)            ;SELECT SUBTEST 2 IN FW
4107 036350  005037  002364                    CLR     NXMTST                  ;CLEAR NXM FLAG
4108 036354  004737  036610                    CALL    DMAOSB                  ;CALL COMMON SUBROUTINE
4109 036360  103443                            BCS     20$                     ;ERROR
4110 036362                                    SAVE    1
4111 036366  012701  002633                    MOV     @TRXBUF+1,R1            ;GET EXPECTED BUFFER ADDRESS
4112 036372  012702  004626                    MOV     @RCVBUF,R2              ;GET RECEIVE BUFFER ADDRESS
4113 036376  012737  001773  002470            MOV     @1019.,BUFLEN           ;LOAD BYTE COUNT
4114 036404  004737  010444                    CALL    COMPRB                  ;COMPARE BOTH BUFFERS BYTES PER BYTE
4115 036410  103015                            BCC     15$                     ;SUCCESS
4116 036412  012737  000020  002470            MOV     @CSRLEN,BUFLEN          ;REINIT BUFLEN
4117 036420  012703  004625                    MOV     @TRXBUF+1019.,R3        ;LOAD END OF EXPECTED RCV BUFFER
4118 036424                                    ERRHRD  60,EM0060,PRDMAR        ;DATA COMPARE ERROR
4119 036434                                    GET     1
4120 036440                                    ESCAPE TST
4121 036444                             15$:   GET     1
4122 036450  022737  000007  002362            CMP     @7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4123 036456  001004                            BNE     20$                     ;NO, QUIT THIS IS THE END
4124 036460  005721                            TST     (R1)+                   ;NEXT SPEED
4125 036462  022701  000016                    CMP     @16,R1                  ;END OF TABLE ?
4126 036466  001276                            BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4127 036470  012737  000020  002470    20$:    MOV     @CSRLEN,BUFLEN          ;REINIT BUFLEN
4128 036476  000240                            NOP
4129 036500                                    ENDTST
4130 036502                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST28 **
                                        ;
                                        ;
                                        ;
                                        ;
4131                                    ;
4132                                    ; NON EXISTENT MEMORY TEST DURING RECEIVE DMA TRANSFER
4133                                    ;
4134                                    ;--
4135 036502                             BGNTST
4136 036502  012737  000001  002364            MOV     #1,NXMTST               ;SET FLAG TO MARK NXM TEST
4137 036510  004737  041542                    JSR     PC,$SIZE                ;GET A REAL NON EXISTENT MEMORY ADDRESS
4138 036514  004737  034626                    CALL    STRSB0                  ;START SUBROUTINE 0
```

```
4139 036520  004737  034646          1$:     CALL    STRSB1                  ;START SUBROUTINE 1
4140 036524  013765  002526  000006          MOV     NXMLOW,SEL6(R5)         ;LOAD LOW NXM ADDRESS /V01.04/
4141 036532  013765  002530  000010          MOV     NXMHIG,SEL10(R5)        ;LOAD HIGH NXM ADDRESS /V01.04/
4142 036540  012765  000012  000004          MOV     #10.,SEL4(R5)           ;LOAD BYTE COUNT
4143 036546  005465  000004                  NEG     SEL4(R5)                ;GET IT IN 2'S COMPL. FORM
4144 036552  005065  000012                  CLR     SEL12(R5)               ;SUBTEST 1 SELECTED IN FW
4145 036556  004737  036610                  CALL    DMAOSB                  ;CALL COMMON SUBROUTINE
4146 036562  103410                          BCS     20$                     ;UNEXPECTED RESULT
4147 036564  022737  000007  002362          CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4148 036572  001004                          BNE     20$                     ;NO, QUIT THIS IS THE END
4149 036574  005721                          TST     (R1)+                   ;NEXT SPEED
4150 036576  022701  000016                  CMP     #16,R1                  ;END OF TABLE ?
4151 036602  001346                          BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4152 036604  000240                  20$:    NOP
4153 036606                                  ENDTST
4154
4155 036610                          DMAOSB: TESTNB  21                      ;LOAD KMV ROUTINE NUMBER
4156
4157 036616                                  WAIT    1,1000.                 ;WAIT
4158 036626  000512                          BR      20$                     ;SEL0 = 0, CORRECT
4159 036630  000406                          BR      2$                      ;SEL0 = 100, ERROR
4160
4161 036632                                  ERRHRD  11,EM0011               ;IF NO RESPONSE
4162 036642  000261                  11$:    SEC
4163 036644  000207                          RETURN
4164
4165 036646  005737  002364          2$:     TST     NXMTST                  ; LOOK FOR NXM TEST
4166 036652  001404                          BEQ     21$                     ; NO DISPATCH
4167 036654  122765  000012  000002          CMPB    #12,SEL2(R5)            ; REAL NXM ?
4168 036662  001074                          BNE     20$
4169
4170 036664                          21$:    DSPACH  8.
4171 036672  000407                          BR      3$                      ;SEL2=2, LINE COUNTER ERROR
4172 036674  000413                          BR      4$                      ;SEL2=4, TRX/RCV ERROR
4173 036676  000417                          BR      5$                      ;SEL2=6, UNEXPECTED TIMEOUT DURING TRX DMA
4174 036700  000423                          BR      6$                      ;SEL2=10, BYTE COUNT OVERFLOW RCV DMA
4175 036702  000427                          BR      7$                      ;SEL2=12, TIMEOUT DURING RCV DMA
4176 036704  000436                          BR      8$                      ;SEL2=14, UNEXPECTED STATUS RECEIVED
4177 036706  000450                          BR      9$                      ;SEL2=16, INTERRUPT AT ILLEGAL VECTOR
4178 036710  000454                          BR      12$                     ;SEL2=20, UNDERRUN INTERRUPT OCCURS
4179
4180 036712                          3$:     ERRHRD  32,EM0032,PRRATE
4181 036722  000747                          BR      11$
4182
4183 036724                          4$:     ERRHRD  45,EM0044,PDMAR3
4184 036734  000742                          BR      11$
4185
4186 036736                          5$:     ERRHRD  61,EM0061,PDMAR1
4187 036746  000735                          BR      11$
4188
4189 036750                          6$:     ERRHRD  62,EM0062,PDMAR1
4190 036760  000730                          BR      11$
4191
4192 036762  005737  002364          7$:     TST     NXMTST                  ;NXM TEST RUNNING ?
4193 036766  001042                          BNE     10$                     ;YES, CORRECT TIMEOUT EXPECTED
4194 036770                                  ERRHRD  63,EM0063,PDMAR1
4195 037000  000720                          BR      11$
```

```
4196
4197 037002  016537  000004  002412  8$:     MOV     SEL4(R5),BDDAT          ;LOAD RECEIVED STATUS&DATA
4198 037010  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;LOAD EXPECTED STATUS&DATA
4199 037016                                  ERRHRD  46,EM0045,PDMAR2
4200 037026  000705                          BR      11$
4201
4202 037030                          9$:     ERRHRD  51,EM0050,PDMAR3
4203 037040  000700                          BR      11$
4204
4205 037042                          12$:    ERRHRD  35,EM0034,PDMAR3
4206 037052  000673                          BR      11$
4207
4208 037054  005737  002364          20$:    TST     NXMTST                  ;NXM TEST RUNNING ?
4209 037060  001405                          BEQ     10$                     ;NO, REAL SUCCESS
4210 037062                                  ERRHRD  65,EM0065
4211 037072  000663                          BR      11$
4212 037074  000241                  10$:    CLC
4213 037076  000207                          RETURN
4214
4215                                 .EVEN
4216
```

```
4218 037100                     BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST29 **
                                ;
                                ;
                                ;
                                ;
4219                            ; TEST OF DMA TRANSFER IN BOTH DIRECTIONS
4220                            ;
4221                            ; DESCRIPTION:  THE HOST ASKS THE  DCT11 FOR  INITIALIZING A
4222                            ;               DMA TRANSFER IN BOTH DIRECTIONS .
4223                            ;               IN ORDER TO RUN TEST, THE HOST PASSES TO THE
4224                            ;               DCT11 PARAMETERS, SUCH AS:
4225                            ;                       SEL2 = LINE CLOCK RATE
4226                            ;                       SEL4 = DMA TRX BYTE COUNT(2'S COMP.)
4227                            ;                       SEL6 = DMA TRX BUS ADDRESS(EVEN)
4228                            ;                       SEL10= DMA TRX IN EXT. ADDRESS
4229                            ;                       SEL12= DMA RCV BYTE COUNT
4230                            ;                       SEL14= DMA RCV OUT BUS ADDRESS
4231                            ;                       SEL16= DMA RCV OUT EXT. ADDRESS
4232                            ;                          BIT15 OF SEL16 IS SET IF SEL20 IS NOT VALID
4233                            ;                          THAT MEANS THE DATA BYTE MUST BE IGNORED WHEN
4234                            ;                          A STATUS IS RECEIVED.
4235                            ;                       SEL20= LAST EXPECTED DATA TO BE RECEIVED
4236                            ;
4237                            ;               THEN THE HOST PERIODICALLY POLLS  SEL0 FOR A
4238                            ;               RESPONSE FROM THE DCT11:
4239                            ;                       SEL0 = 0 IF TEST COMPLETED
4240                            ;                       SEL0 = 100 IF ANY ERROR
4241                            ;               IN THE FORMER CASE THE HOST COMPARES THE XMT
4242                            ;               TABLE WITH THE RCV ONE AND REPORT  ERROR  64
4243                            ;               IF ANY DATA COMPARE ERROR.
4244                            ;               IN THE LATTER CASE:
4245                            ;                       SEL2 = 2        FOR ERROR 32
4246                            ;                            = 4        FOR ERROR 45
4247                            ;                            = 6        FOR ERROR 56
4248                            ;                            = 10       FOR ERROR 62
4249                            ;                            = 12       FOR ERROR 63
4250                            ;                            = 14       FOR ERROR 55
4251                            ;                            = 16       FOR ERROR 46
4252                            ;                            = 20       FOR ERROR 51
4253                            ;                            = 22       FOR ERROR 35
4254                            ;
4255                            ;                       SEL4 = RECEIVED STATUS&DATA REGISTER (ERROR 46)
4256                            ;                       SEL6 = EXPECTED STATUS&DATA REGISTER (ERROR 46)
4257                            ;                       SEL10= FLAGWORD
4258                            ;                       SEL12= TRX DMA BYTE COUNT
4259                            ;                       SEL14= TRX DMA BUS ADDRESS
4260                            ;                       SEL16= TRX DMA EXT. ADDRESS
4261                            ;                       SEL20= RCV DMA BYTE COUNT
4262                            ;                       SEL22= RCV DMA BUS ADDRESS
4263                            ;                       SEL24= RCV DMA EXT. ADDRESS
4264                            ;               THEN THE HOST BUILDS REPORT AND EXIT
```

```
                                ;
                                ; MAINT MODE:   2
                                ;
                                ; KMV ROUTINE:  22
                                ;
                                ; REPORTS:      ERROR 2     MASTER CLEAR FAILS TO RESET
                                ;               ERROR 11    NO KMV11 RESPONSE
                                ;               ERROR 32    LINE COUNTER ERROR WHEN STARTED
                                ;               ERROR 35    UNDERRUN INTERRUPT OCCURS
                                ;               ERROR 45    TRX/RCV ERRORS
                                ;               ERROR 46    UNEXPECTED STATUS RECEIVED
                                ;               ERROR 51    INTERRUPT AT ILLEGAL VECTOR
                                ;               ERROR 55    BIT "BCNTIN" NOT SET AT END OF TRX DMA
                                ;               ERROR 56    Q-BUS TIMEOUT DURING TRX DMA
                                ;               ERROR 62    BYTE COUNT OVERFLOW DURING RCV DMA
                                ;               ERROR 63    Q-BUS TIMEOUT DURING RCV DMA
                                ;               ERROR 64    DATA COMPARE ERROR
                                ;
                                ;--
 000017                                RXLNG1=15.
 000010                                RXLNG2=8.
 000006                                RXLNG3=6.
 001774                                RXLNG4=1020.
 000013                                TXLNG1=11.
 000012                                TXLNG2=10.
 001774                                TXLNG4=1020.

037100                          BGNTST
                                ;
                                ; THE FIRST TEST OF THIS CLASS CORRESPONDS TO A 11 BYTES LONG TRANSMIT BUFFER
                                ; AND A 15 BYTES LONG RECEIVE BUFFER.
                                ;
037100  004737  034626                1$:     CALL    STRSB0                  ;CALL START SUBROUTINE 0
037104  004737  034646                        CALL    STRSB1                  ;CALL START SUBROUTINE 1
037110  012702  000017                        MOV     #RXLNG1,R2              ;GET LENGTH OF RECEIVE BUFFER
037114  012703  000013                        MOV     #TXLNG1,R3              ;LOAD TRANSMIT BUFFER LENGTH
037120  004737  037350                        CALL    DMASUB                  ;CALL COMMON SUBROUTINE
037124  103410                              BCS     20$                     ;ERROR
037126  022737  000007  002362              CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
037134  001004                              BNE     20$                     ;NO, QUIT THIS IS THE END
037136  005721                              TST     (R1)+                   ;NEXT SPEED
037140  022701  000016                      CMP     #16,R1                  ;END OF TABLE ?
037144  001357                              BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
037146  000240                        20$:  NOP
037150                          ENDTST
                                .EVEN
037152                          BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST30 **
                                ;
                                ;
                                ;
                                ;
```

```
4312                                    ;
4313                                    ; THE SECOND TEST OF THIS CLASS CORRESPONDS TO A 10 BYTES LONG TRANSMIT BUFFER
4314                                    ; AND A 8 BYTES LONG RECEIVE BUFFER.
4315                                    ;--
4316 037152                             BGNTST
4317 037152  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4318 037156  004737  034646             1$:    CALL    STRSB1                  ;CALL START SUBROUTINE 1
4319 037162  012702  000010                    MOV     #RXLNG2,R2              ;LOAD RECEIVE BUFFER LENGTH
4320 037166  012703  000012                    MOV     #TXLNG2,R3              ;LOAD TRANSMIT BUFFER LENGTH
4321 037172  004737  037350                    CALL    DMASUB                  ;CALL COMMON SUBROUTINE
4322 037176  103410                           BCS     20$                     ;ERROR
4323 037200  022737  000007  002362           CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4324 037206  001004                           BNE     20$                     ;NO, QUIT THIS IS THE END
4325 037210  005721                           TST     (R1)+                   ;NEXT SPEED
4326 037212  022701  000016                    CMP     #16,R1                  ;END OF TABLE ?
4327 037216  001357                           BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4328 037220  000240                    20$:   NOP
4329 037222                             ENDTST
4330                                    .EVEN
4331 037224                             BADHEAD
                                        ;++
                                        ;
                                        ;
                                        ;
                                        ;                       ++ TEST31 ++
                                        ;
                                        ;
                                        ;
4332                                    ;
4333                                    ; THE THIRD TEST OF THIS CLASS CORRESPONDS TO A 10 BYTES LONG TRANSMIT BUFFER
4334                                    ; AND A 6 BYTES LONG RECEIVE BUFFER. THIS TEST PERMITS TO TEST THE BUFFER
4335                                    ; OVERFLOW.
4336                                    ;--
4337 037224                             BGNTST
4338 037224  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4339 037230  004737  034646             1$:    CALL    STRSB1                  ;CALL START SUBROUTINE 1
4340 037234  012702  000006                    MOV     #RXLNG3,R2              ;LOAD RECEIVE BUFFER LENGTH
4341 037240  012703  000012                    MOV     #TXLNG2,R3              ;LOAD TRANSMIT BUFFER LENGTH
4342 037244  004737  037350                    CALL    DMASUB                  ;CALL COMMON SUBROUTINE
4343 037250  103410                           BCS     20$                     ;ERROR
4344 037252  022737  000007  002362           CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4345 037260  001004                           BNE     20$                     ;NO, QUIT THIS IS THE END
4346 037262  005721                           TST     (R1)+                   ;NEXT SPEED
4347 037264  022701  000016                    CMP     #16,R1                  ;END OF TABLE ?
4348 037270  001357                           BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4349 037272  000240                    20$:   NOP
4350 037274                             ENDTST
4351                                    .EVEN
4352 037276                             BADHEAD
                                        ;++
                                        ;
                                        ;
                                        ;
                                        ;                       ++ TEST32 ++
```

```
                                        ;
                                        ;
                                        ;
                                        ;
4353                                    ;
4354                                    ; THE FOURTH TEST OF THIS CLASS CORRESPONDS TO A LONG DMA TRANSFER.
4355                                    ; 1020 BYTES WILL BE TRANSMITTED AND RECEIVED.
4356                                    ;--
4357 037276                             BGNTST
4358 037276  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4359 037302  004737  034646             1$:    CALL    STRSB1                  ;CALL START SUBROUTINE 1
4360 037306  012702  001774                    MOV     #RXLNG4,R2              ;LOAD RECEIVE BUFFER LENGTH
4361 037312  012703  001774                    MOV     #TXLNG4,R3              ;LOAD TRANSMIT BUFFER LENGTH
4362 037316  004737  037350                    CALL    DMASUB                  ;CALL COMMON SUBROUTINE
4363 037322  103410                           BCS     20$                     ;ERROR
4364 037324  022737  000007  002362           CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4365 037332  001004                           BNE     20$                     ;NO, QUIT THIS IS THE END
4366 037334  005721                           TST     (R1)+                   ;NEXT SPEED
4367 037336  022701  000016                   CMP     #16,R1                  ;END OF TABLE ?
4368 037342  001357                           BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4369 037344  000240                    20$:   NOP
4370 037346                             ENDTST
4371                                    .EVEN
4372                                    ;
4373                                    ; COMMON SUBROUTINE FOR DMA TESTS IN BOTH DIRECTIONS
4374                                    ; IT ACTIVATES FIRMWARE ROUTINE TEST22 AFTER HAVING LOADED THE CSR WITH
4375                                    ; THE CORRESPONDING PARAMETERS.
4376                                    ;
4377                                    ; INPUT: R2 CONTAINS THE RECEIVE BUFFER LENGTH (IN BYTES)
4378                                    ;        R3 CONTAINS THE TRANSMIT BUFFER LENGTH (IN BYTES)
4379                                    ;
4380 037350                             DMASUB:
4381 037350                                    SAVE    1
4382 037354  012701  004626                    MOV     #RCVBUF,R1              ;GET RECEIVE BUFFER
4383 037360  006202                           ASR     R2                      ;GET BYTE COUNT IN WORDS
4384 037362  010237  002470                    MOV     R2,BUFLEN               ;LOAD CORRESPONDING LENGTH
4385 037366  006302                           ASL     R2                      ;RESTORE R2
4386 037370  004737  011052                    CALL    CLEAR                   ;CLEAR THE RECEIVE BUFFER
4387 037374                                    GET     1
4388 037400  012765  004626  000014           MOV     #RCVBUF,SEL14(R5)       ;LOAD RECEIVE BUFFER ADDRESS
4389 037406  005065  000016                    CLR     SEL16(R5)               ;LOAD EXT. RECEIVE BUFFER ADDRESS
4390 037412  010265  000012                    MOV     R2,SEL12(R5)            ;LOAD RECEIVE BUFFER LENGTH
4391 037416  005465  000012                    NEG     SEL12(R5)               ;GET IT IN 2'S COMPLEMENT
4392 037422  012765  002632  000006           MOV     #TRXBUF,SEL6(R5)        ;LOAD TRANSMIT BUFFER LENGTH
4393 037430  005065  000010                    CLR     SEL10(R5)               ;LOAD EXT. TRANSMIT BUFFER ADDRESS
4394 037434  010365  000004                    MOV     R3,SEL4(R5)             ;LOAD TRANSMIT BUFFER LENGTH
4395 037440  005465  000004                    NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT
4396 037444  062703  002631                    ADD     #TRXBUF-1,R3            ;GET LAST TRX DATA ADDRESS
4397 037450  111365  000020                    MOVB    (R3),SEL20(R5)          ;LOAD LAST DATA IN SEL20
4398 037454  162703  002631                    SUB     #TRXBUF-1,R3            ;RESTORE R3
4399 037460                                    TESTNB  22                      ;ACTIVATE TEST22 IN FIRMWARE
4400
4401 037466                                    WAIT    1,1000.                 ;WAIT
4402 037476  000517                           BR      20$                     ;SEL0=0 ,SUCCESS
4403 037500  000411                           BR      2$                      ;SEL0=100 , ERROR
4404 037502                                    ERRHRD  11,EM0011               ;TIMEOUT, NO KMV11-C RESPONSE
4405 037512  012737  000020  002470  1$:      MOV     #CSRLEN,BUFLEN          ;RESTORE CSR LENGTH
```

```
4406 037520  000261                        SEC
4407 037522  000207                        RETURN
4408
4409 037524                          2$:   DSPACH  9.                ;DISPATCH FOLLOWING SEL2 VALUE
4410 037532  000410                        BR      3$                ;SEL2=2 LINE COUNTER ERROR
4411 037534  000414                        BR      4$                ;SEL2=4 TRX/RCV ERRORS
4412 037536  000420                        BR      5$                ;SEL2=6 Q-BUS TIMEOUT DURING TRX DMA
4413 037540  000424                        BR      6$                ;SEL2=10 BYTE COUNT OVERFLOW DURING RCV DMA
4414 037542  000436                        BR      7$                ;SEL2=12 Q-BUS TIMEOUT DURING RCV DMA
4415 037544  000442                        BR      8$                ;SEL2=14 BIT "BCNTIN" NOT SET AT END OF TRX DMA
4416 037546  000446                        BR      9$                ;SEL2=16 UNEXPECTED STATUS RECEIVED
4417 037550  000460                        BR      10$               ;SEL2=20 INTERRUPT AT ILLEGAL VECTOR
4418 037552  000464                        BR      11$               ;SEL2=22 UNDERRUN INTERRUPT OCCURS
4419
4420 037554                          3$:   ERRHRD  32,EM0032,PRRATE
4421 037564  000752                        BR      1$
4422
4423
4424 037566                          4$:   ERRHRD  45,EM0044,PDMAB2
4425 037576  000745                        BR      1$
4426
4427 037600                          5$:   ERRHRD  56,EM0056,PDMART
4428 037610  000740                        BR      1$
4429
4430 037612  020203                  6$:   CMP     R2,R3             ;RCV LENGTH > TRX LENGTH ?
4431 037614  003004                        BGT     22$               ;YES, REAL ERROR
4432 037616  012737  000020  002470        MOV     #CSRLEN,BUFLEN    ;RESTORE CSR LENGTH
4433 037624  000207                        RETURN                    ;EXPECTED BYTE COUNT OVERFLOW
4434 037626                         22$:   ERRHRD  62,EM0062,PDMART
4435 037636  000725                        BR      1$
4436
4437 037640                          7$:   ERRHRD  63,EM0063,PDMART
4438 037650  000720                        BR      1$
4439
4440 037652                          8$:   ERRHRD  55,EM0055,PDMART
4441 037662  000713                        BR      1$
4442
4443 037664  016537  000004  002412  9$:   MOV     SEL4(R5),BDDAT    ;LOAD RECEIVED STATUS&DATA
4444 037672  016537  000006  002410        MOV     SEL6(R5),GDDAT    ;LOAD EXPECTED STATUS&DATA
4445 037700                                ERRHRD  46,EM0045,PDMAB1
4446 037710  000700                        BR      1$
4447
4448 037712                         10$:   ERRHRD  51,EM0050,PDMAB2
4449 037722  000673                        BR      1$
4450
4451 037724                         11$:   ERRHRD  35,EM0034,PDMAB2
4452 037734  000666                        BR      1$
4453
4454 037736                         20$:   SAVE    1
4455 037742  010337  002470                MOV     R3,BUFLEN         ;LOAD BUFFER LENGTH TO CHECK FOR
4456                                                                  ;COMPAR SUBROUTINE
4457 037746  162737  000001  002470        SUB     #1,BUFLEN         ;MINUS 1 FOR 1ST DATA
4458                                                                  ;RECEIVED IN INTERRUPT MODE
4459 037754  012701  002633                MOV     #TRXBUF+1,R1      ;LOAD SOURCE BUFFER ADDRESS
4460 037760  012702  004626                MOV     #RCVBUF,R2        ;LOAD RECEIVE BUFFER ADDRESS
4461 037764  004737  010444                CALL    COMPRB            ;COMPARE BOTH BYTE PER BYTE
4462 037770  103406                        BCS     25$               ;ERROR
```

```
4463 037772                                    GET     1
4464 037776  012737  000020  002470            MOV     #CSRLEN,BUFLEN          ;RESTORE CSR LENGTH IN BUFLEN
4465 040004  000207                            RETURN
4466 040006  012737  000020  002470  25$:      MOV     #CSRLEN,BUFLEN          ;RESTORE CSR LENGTH IN BUFLEN
4467 040014  062703  002631                    ADD     #TRXBUF-1,R3            ;GET END  TRANSMIT BUFFER ADDRESS
4468 040020                                    ERRHRD  64,EM0064,PRDMAR        ;"DATA COMPARE ERROR"
4469 040030                                    GET     1
4470 040034  000261                            SEC
4471 040036  000207                            RETURN
```

```
4473 040040                            BADHEAD
                                       ;**
                                       ;
                                       ;
                                       ;
                                       ;                       ** TEST33 **
                                       ;
                                       ;
                                       ;
4474                                   ; TEST OF DMA TRANSFER IN BOTH DIRECTIONS AND IN THE HIGHER PART OF THE
4475                                   ; HOST MEMORY TO CHECK THE EXTENDED ADDRESS COUNTER
4476                                   ;
4477                                   ; INPUT PARAMETERS FOR FW ROUTINE:
4478                                   ;       SEL2 = LINE CLOCK RATE
4479                                   ;       SEL4 = TRX DMA BYTE COUNT (IN 2'S COMP FORM)
4480                                   ;       SEL6 = TRX DMA BUFFER ADDRESS (LOW)
4481                                   ;       SEL10= TRX DMA BUFFER ADDRESS (HIGH)
4482                                   ;       SEL12= RCV DMA BYTE COUNT (IN 2'S COMP FORM)
4483                                   ;       SEL14= RCV DMA BUFFER ADDRESS (LOW)
4484                                   ;       SEL16= RCV DMA BUFFER ADDRESS (HIGH)
4485                                   ;              BIT 15 SET IF SEL20 MUST NOT BE TAKEN INTO ACCOUNT IN FW TEST22
4486                                   ;       SEL20= LAST EXPECTED DATA TO BE RECEIVED
4487                                   ;
4488                                   ; OUTPUT:
4489                                   ;       SEL0 = 0 IF SUCCESS
4490                                   ;       SEL0 = 100 IF ANY ERROR
4491                                   ;                WITH SEL2 =
4492                                   ;
4493                                   ;                    SEL12= TRX DMA BYTE COUNT
4494                                   ;                    SEL14= TRX DMA BUFFER ADDRESS (LOW)
4495                                   ;                    SEL16= TRX EXT ADDRESS
4496                                   ;                    SEL20= RCV DMA BYTE COUNT
4497                                   ;                    SEL22= RCV DMA BUFFER ADDRESS (HIGH)
4498                                   ;                    SEL24= RCV DMA EXT. BUFFER ADDRESS
4499                                   ;
4500                                   ; MAINT MODE:   2
4501                                   ;
4502                                   ; KMV ROUTINE:  22
4503                                   ;
4504                                   ;--
4505          000000                           TRXBFH = 0                    ;TRX BUFFER ADDRESS (BITS 16-21)
4506          107700                           TRXBFL = 107700               ;TRX BUFFER ADDRESS (BITS 0-15) /V01.05/
4507          001000                           TRXLNG = 1000                 ;TRX BUFFER LENGTH
4508          001000                           RCVLNG = 1000                 ;RCV BUFFER LENGTH
4509
4510 040040  000000                           RCVBFH: .WORD 0               ;RCV BUFFER ADDRESS (BITS 16-21) /V01.05/
4511 040042  000000                           RCVBFL: .WORD 0               ;RCV BUFFER ADDRESS (BITS 0-15) /V01.05/
4512 040044                            BGNTST
4513 040044  005037  002364                      CLR     NXMTST            ;CLEAR NXM FLAG
4514 040050  004737  041542                      JSR     PC,$SIZE          ;CALL SUBROUTINE SIZE TO GET MAX ADDRESS
4515                                                                        ;VALUE IN THE HOST MEMORY
4516 040054  005737  042070                      TST     $LSTBK            ;CHECK EXTENDED ADDRESS RETURNED BY $SIZE
4517 040060  001002                              BNE     30$               ;IF IT IS NULL , NO MORE 16-BITS IN MEMORY
4518 040062  000137  040506                      JMP     ENDT33            ;CANNOT CHECK EXTENDED ADDRESS,QUIT
4519                                                                        ; DO THE TEST OTHERWISE
```

```
4520 040066                                30$:    SAVE    4                       ; /V01.05/
4521 040072  013704  042066                       MOV     $LSTAD,R4               ; CALCULATE RX BUF STARTING ADDRESS /V01.05/
4522 040076  162704  001000                       SUB     @RCVLNG,R4              ; /V01.05/
4523 040102  162704  001000                       SUB     @RCVLNG,R4              ; /V01.05/
4524 040106  010437  040042                       MOV     R4,RCVBFL               ; /V01.05/ LOAD BITS 0 TO 15
4525 040112  013737  042070  040040               MOV     $LSTBK,RCVBFH           ; /V01.05/ LOAD BITS 16 TO 21
4526 040120                                        GET     4                       ; /V01.05/
4527
4528 040124  004737  034626                1$:     CALL    STRSB0
4529 040130  004737  034646                        CALL    STRSB1
4530 040134  005037  107700                        CLR     @#TRXBFL                ; /V01.05/ FIRT TX CHAR MUST BE ZERO
4531 040140  012765  001000  000004               MOV     @TRXLNG,SEL4(R5)        ;LOAD TRX BYTE COUNT
4532 040146  005465  000004                        NEG     SEL4(R5)                ;GET IT IN 2'S COMPLEMENT
4533 040152  012765  107700  000006               MOV     @TRXBFL,SEL6(R5)        ;LOAD TRX BUFFER ADDRESS
4534 040160  012765  000000  000010               MOV     @TRXBFH,SEL10(R5)       ;LOAD EXTENDED ADDRESS
4535 040166  012765  001000  000012               MOV     @RCVLNG,SEL12(R5)       ;LOAD RCV BYTE COUNT
4536 040174  005465  000012                        NEG     SEL12(R5)               ;GET IT IN 2'S COMPLEMENT
4537 040200  013765  040042  000014               MOV     RCVBFL,SEL14(R5)        ;LOAD RCV BUFFER ADDRESS (LOW) /V01.05/
4538 040206  013765  040040  000016               MOV     RCVBFH,SEL16(R5)        ;LOAD RCV EXTENDED ADDRESS /V01.05/
4539 040214  052765  100000  000016               BIS     #100000,SEL16(R5)       ;SET FLAG FOR TEST22 FW ROUTINE
4540 040222                                        TESTNB  22                      ;ACTIVATE FW ROUTINE 22
4541 040230                                        WAIT    1,1000.                 ;WAIT
4542 040240  000512                                BR      20$                     ;SEL0=0 ,SUCCESS
4543 040242  000406                                BR      2$                      ;SEL0=100 , ERROR
4544 040244                                        ERRHRD  11,EM0011               ;TIMEOUT, NO KMV11-C RESPONSE
4545 040254                                        ESCAPE  TST
4546
4547 040260                                2$:     DSPACH  9.                      ;DISPATCH FOLLOWING SEL2 VALUE
4548 040266  000410                                BR      3$                      ;SEL2=2 LINE COUNTER ERROR
4549 040270  000415                                BR      4$                      ;SEL2=4 TRX/RCV ERRORS
4550 040272  000422                                BR      5$                      ;SEL2=6 Q-BUS TIMEOUT DURING TRX DMA
4551 040274  000427                                BR      6$                      ;SEL2=10 BYTE COUNT OVERFLOW DURING RCV DMA
4552 040276  000427                                BR      7$                      ;SEL2=12 Q-BUS TIMEOUT DURING RCV DMA
4553 040300  000434                                BR      8$                      ;SEL2=14 BIT "BCNTIN" NOT SET AT END OF TRX DMA
4554 040302  000441                                BR      9$                      ;SEL2=16 UNEXPECTED STATUS RECEIVED
4555 040304  000454                                BR      10$                     ;SEL2=20 INTERRUPT AT ILLEGAL VECTOR
4556 040306  000461                                BR      11$                     ;SEL2=22 UNDERRUN INTERRUPT OCCURS
4557
4558 040310                                3$:     ERRHRD  32,EM0032,PRRATE
4559 040320                                        ESCAPE  TST
4560
4561
4562 040324                                4$:     ERRHRD  45,EM0044,PDMAB2
4563 040334                                        ESCAPE  TST
4564
4565 040340                                5$:     ERRHRD  56,EM0056,PDMART
4566 040350                                        ESCAPE  TST
4567
4568 040354  000444                        6$:     BR      20$                     ;RCV AND TRX LENGTH ARE EQUAL
4569
4570
4571 040356                                7$:     ERRHRD  63,EM0063,PDMART
4572 040366                                        ESCAPE  TST
4573
4574 040372                                8$:     ERRHRD  55,EM0055,PDMART
4575 040402                                        ESCAPE  TST
4576
```

```
4577 040406  016537  000004  002412  9$:     MOV     SEL4(R5),BDDAT          ;LOAD RECEIVED STATUS&DATA
4578 040414  016537  000006  002410          MOV     SEL6(R5),GDDAT          ;LOAD EXPECTED STATUS&DATA
4579 040422                                  ERRHRD  46,EM0045,PDMAB1
4580 040432                                  ESCAPE  TST
4581
4582 040436                          10$:    ERRHRD  51,EM0050,PDMAB2
4583 040446                                  ESCAPE  TST
4584
4585 040452                          11$:    ERRHRD  35,EM0034,PDMAB2
4586 040462                                  ESCAPE  TST
4587
4588 040466                          20$:
4589 040466  022737  000007  002362          CMP     #7,RATE                 ;WAS IT ALL LINE RATE SELECTED ?
4590 040474  001004                          BNE     ENDT33                  ;NO, QUIT THIS IS THE END
4591 040476  005721                          TST     (R1)+                   ;NEXT SPEED
4592 040500  022701  000016                  CMP     #16,R1                  ;END OF TABLE ?
4593 040504  001211                          BNE     1$                      ;NO CONTINUE WITH NEXT LINE RATE
4594 040506                          ENDT33:
4595 040506  000240                          NOP
4596 040510                          ENDTST
4597
```

```
4599 040512                             BADHEAD
                                        ;++
                                        ;
                                        ;
                                        ;
                                        ;                       ++ TEST34 ++
                                        ;
                                        ;
                                        ;
4600                                    ; TEST OF MODEM SIGNALS (EXTERNAL LOOP-BACK MODE)
4601                                    ;
4602                                    ; DESCRIPTION:  THIS TEST CAN BE RUN IN THE CASE OF EXTERNAL
4603                                    ;               LOOP ONLY. AN ERROR MESSAGE IS PRINTED IF NO
4604                                    ;               LOOP-BACK CONNECTOR PLUGGED.
4605                                    ;               OTHERWISE, THE HOST ASKS THE DCT11 TO SET-UP
4606                                    ;               MODEM CONTROLS BY MEANS OF THE 'MAILBOX' ROU
4607                                    ;               TINE WITH THE FOLLOWING INPUTS:
4608                                    ;                       SEL2 = PORT B ADDRESS
4609                                    ;                       SEL4 = MODEM CONTROL BIT(S)
4610                                    ;                       SEL6 = 6 FOR WRITE BYTE
4611                                    ;
4612                                    ;               THEN TO READ LOOP BACK RESULT BY MEANS OF THE
4613                                    ;               'MAILBOX ROUTINE WITH THE FOLLOWING INPUTS:
4614                                    ;                       SEL2 = PORT A ADDRESS
4615                                    ;                       SEL4 = 377
4616                                    ;                       SEL6 = 4 FOR READ BYTE
4617                                    ;               THE HOST POLLS SEL0 FOR "MAILBOX" ROUTINE JOB
4618                                    ;               COMPLETED:
4619                                    ;                       SEL0 = 0 FOR JOB COMPLETE
4620                                    ;               IN THE LOOP BACK RESULT CASE,
4621                                    ;                       SEL4 = EXPCTED LOOP BACK RESULT
4622                                    ;               THEN THE HOST TESTS/REPORTS AND EXITS.
4623                                    ;
4624                                    ;               NOTICE, THAT MODEM SIGNALS  ARE LOOPED, SUCH
4625                                    ;               AS:
4626                                    ;
4627                                    ;               CCITT 108/2 --------.
4628                                    ;                                   )
4629                                    ;               CCITT 105   ----.   )
4630                                    ;                               )   )
4631                                    ;               CCITT 107   <---)---'
4632                                    ;                               )
4633                                    ;               CCITT 106   <---)
4634                                    ;                               )
4635                                    ;               CCITT 109   <---'
4636                                    ;
4637                                    ;               MODEM CCITT 108/2 AND 105 WILL BE SET ONE BY
4638                                    ;               ONE THEN TOGETHER.
4639                                    ;
4640                                    ; MAINT MODE:   2
4641                                    ;
4642                                    ; KMV ROUTINE:  23
4643                                    ;
4644                                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
4645                                    ;               ERROR 11        NO KMV11 RESPONSE
```

```
                                        ;               ERROR 74        MODEM SIGNAL BIT SETTING NOT
                                        ;                               CORRECT
                                        ;               ERROR 75        NO LOOP BACK CONNECTOR
                                        ;--

040512                                  BGNTST
040512  004737  011116                          CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS
040516                                          MODE    2                       ;SELECT MAINTENANCE MODE 1
040522  103002                                  BCC     1000$
040524                                          ESCAPE  TST
040530                                  1000$:
040530  005737  002366                          TST     LCLOOP                  ;EXTERNAL LOOP BACK POSSIBLE?
040534  001012                                  BNE     1$                      ;IF YES

040536                                          PRINTF  #EM0052                 ;IF NOT, REPORT /V01.04/
040556  000137  041070                          JMP     T23A01                  ; /V01.04/

040562                                  1$:

040562                                  BGNSUB
040564  012737  000100  002360                  MOV     #BIT6,DATA              ;SET "DATA TERMINAL READY" (PORTB)
040572  012737  000002  002350                  MOV     #BIT1,GOOD              ;"DATA SET READY" EXPECTED (PORTA)
040600  004737  040676                          CALL    T23A00                  ;TEST
040604                                          ESCAPE  TST
040610                                  ENDSUB

040612                                  BGNSUB
040614  012737  000200  002360                  MOV     #BIT7,DATA              ;SET "REQUEST TO SEND" (PORTB)
040622  012737  000005  002350                  MOV     #BIT0!BIT2,GOOD         ;"CLEAR TO SEND AND CARRIER DETECT"
                                                                                ;EXPECTED (PORTA)
040630  004737  040676                          CALL    T23A00                  ;TEST
040634                                          ESCAPE  TST
040640                                  ENDSUB

040642                                  BGNSUB
040644  012737  000300  002360                  MOV     #BIT6!BIT7,DATA         ;SET BOTH IN PORTB
040652  012737  000007  002350                  MOV     #BIT0!BIT1!BIT2,GOOD    ;RESULT=ALL SET IN PORTA
040660  004737  040676                          CALL    T23A00                  ;TEST
040664                                          ESCAPE  TST
040670                                  ENDSUB
040672                                          EXIT    TST

040676                                  T23A00:
040676  013765  002360  000004                  MOV     DATA,SEL4(R5)           ;SET CORRESPONDING BITS IN SEL4
040704  012765  130012  000002                  MOV     #130012,SEL2(R5)        ;LOAD PORTB ADDRESS
040712  012765  000006  000006                  MOV     #6,SEL6(R5)             ;LOAD "WRITE" CODE FOR TEST23 SUB.
040720                                          TESTNB  23                      ;SELECT TEST23 IN FW
040726                                          WAIT    1.1                     ;WAIT 1 MS AND CHECK SEL0
040736  000406                                  BR      1$                      ;SEL0 IS 0, CORRECT
040740  000240                                  NOP                             ;NO POSSIBLE ERROR
040742                                          ERRHRD  11,EM0011               ;IF NO RESPONSE
040752  000207                                  RETURN

040754  012765  130000  000002  1$:             MOV     #130000,SEL2(R5)        ;LOAD PORT A ADDRESS
040762  012765  000377  000004                  MOV     #377,SEL4(R5)           ;!!!
040770  012765  000004  000006                  MOV     #4,SEL6(R5)             ;LOAD "READ CODE" FOR TEST23 SUB.
```

```
4703 040776                                    TESTNB  23                      ;SELECT TEST23 IN FW
4704 041004                                    WAIT    1,1                     ;WAIT 1 MS AND CHECK SEL0
4705 041014  000406                            BR      2$                      ;SEL0 IS 0, CORRECT
4706 041016  000240                            NOP                             ;NO POSSIBLE ERROR
4707 041020                                    ERRHRD  11,EM0011               ;IF NO RESPONSE
4708 041030  000207                            RETURN
4709
4710 041032  016537  000004  002352  2$:      MOV     SEL4(R5),BAD            ;READ PORT A
4711 041040  042737  177770  002352            BIC     #177770,BAD             ;KEEP ONLY VALID MODEM BITS
4712 041046  023737  002352  002350            CMP     BAD,GOOD                ;EXPECTED RESULT ?
4713 041054  001404                            BEQ     3$                      ;YES
4714 041056                                    ERRHRD  74,EM0051,PRMDM         ;IF ERROR
4715 041066                            3$:
4716 041066  000207                            RETURN
4717 041070                            T23A01:                                 ; /V01.04/
4718 041070                            ENDTST
4719                                   .EVEN
```

```
 4721 041072                    BADHEAD
                                ;**
                                ;
                                ;
                                ;
                                ;                       ** TEST35 **
                                ;
                                ;
                                ;
 4722                           ; THIS TEST LOADS A ROUTINE IN THE RAM AND CHECKS IF ITS
 4723                           ; EXECUTION IS CORRECT.
 4724                           ; THE ROUTINE LOADED IS THE FW TEST06
 4725                           ;
 4726                           ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING ALL R/W
 4727                           ;               REGISTERS BY ITSELF. PERIODICALLY, THE HOST
 4728                           ;               POLLS SEL0 FOR ANY DCT11 RESPONSE:
 4729                           ;                       SEL0 = 0        FOR TEST OKE
 4730                           ;                       SEL0 = 100      FOR ANY ERROR
 4731                           ;               IN THE LATTER CASE, THE FOLLOWING REGISTERS
 4732                           ;               CONTAINS ERROR REPORTS, SUCH AS:
 4733                           ;                       SEL2 = 2        BYTE COUNT OUT REG.
 4734                           ;                            = 4        BUS ADDRESS OUT REG.
 4735                           ;                            = 6        EXT ADDRESS OUT REG.
 4736                           ;                            = 10       BYTE COUNT IN REG.
 4737                           ;                            = 12       BUS ADDRESS IN REG.
 4738                           ;                            = 14       EXT ADDRESS IN REG.
 4739                           ;                       SEL4 = CURRENT PATTERN
 4740                           ;                       SEL6 = BAD WORD
 4741                           ;
 4742                           ; MAINT MODE:   2
 4743                           ;
 4744                           ; KMV ROUTINE:  27
 4745                           ;
 4746                           ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
 4747                           ;               ERROR 11        NO RESPONSE FROM THE KMV11
 4748                           ;               ERROR 17        DATA COMPARE ERROR ON KMV11
 4749                           ;                               INTERNAL R/W REGISTER
 4750                           ;
 4751                           ;
 4752                           ;--
 4753
 4754
 4755 041072                    BGNTST
 4756 041072  004737  011116            CALL    CLRKMV                  ;CLEAN CSR'S AND GET CSR ADDRESS
 4757 041076                            MODE    2                       ;SELECT MAINTENANCE MODE 1
 4758 041102  103002                    BCC     1000$
 4759 041104                            ESCAPE  TST
 4760 041110                    1000$:
 4761 041110                            TESTNB  27                      ;SELECT TEST 6 IN FW
 4762 041116                            WAIT    1,100.                  ;WAIT 100MS AND CHECK SEL0
 4763 041126  000426                    BR      3$                      ;SEL0 IS 0, CORRECT
 4764 041130  000406                    BR      1$                      ;IF ANY ERROR
 4765
 4766 041132                            ERRHRD  11,EM0011               ;IF NO RESPONSE
 4767 041142                            ESCAPE TST
```

```
4768
4769 041146  016537  000002  002352  1$:     MOV     SEL2(R5),BAD            ;GET ERROR CODE
4770 041154  016537  000004  002360          MOV     SEL4(R5),DATA           ;GET PATTERN
4771 041162  016537  000006  002412          MOV     SEL6(R5),BDDAT          ;GET BAD VALUE
4772 041170                                  ERRHRD  17,EM0016,INREG         ;REPORT
4773 041200                                  ESCAPE TST
4774
4775 041204  000240                  3$:     NOP
4776 041206                          ENDTST
4777
```

```
4779 041210                             BADHEAD
                                        ;**
                                        ;
                                        ;
                                        ;
                                        ;                       ** TEST36 **
                                        ;
                                        ;
                                        ;
4780                                    ; RECEIVE DATA TEST USING THE POLLING FACILITY INSTEAD OF INTERRUPT
4781                                    ;
4782                                    ; DESCRIPTION:
4783                                    ;               THIS TEST SELECT IN THE NATIVE FIRMWARE THE TEST30,
4784                                    ;               WHICH TESTS RECEPTION OF DATA USING THE POLLING FACILITY
4785                                    ;               INSTEAD OF INTERRUPT.
4786                                    ;               PERIODICALLY THE HOST POLLS SEL0 FOR ANY DCT11
4787                                    ;               RESPONSE:
4788                                    ;                       SEL0 = 0        FOR TEST OKE
4789                                    ;                       SEL0 = 100      IF ANY ERROR
4790                                    ;               IN THE LATTER CASE SEL2 POINTS TO THE ERROR,
4791                                    ;               POSSIBLE ERRORS ARE :
4792                                    ;                       SEL2 = 2        FOR ERROR 32
4793                                    ;                            = 4        FOR ERROR 45
4794                                    ;                            = 6        FOR ERROR 40
4795                                    ;                            = 10       FOR ERROR 46
4796                                    ;                            = 12       FOR ERROR 47
4797                                    ;                            = 14       FOR ERROR 50
4798                                    ;                            = 16       FOR ERROR 51
4799                                    ;                            = 20       FOR ERROR 35
4800                                    ;                            = 22       FOR ERROR 43
4801                                    ;                   SEL4 = OBTAINED STATUS&DATA
4802                                    ;                   SEL6 = EXPECTED ONE
4803                                    ;                   SEL10= EVENT FLAGS
4804                                    ;
4805                                    ; MAINT MODE:   2
4806                                    ;
4807                                    ; KMV ROUTINE:  30
4808                                    ;
4809                                    ;
4810                                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
4811                                    ;               ERROR 11        NO KMV11 RESPONSE
4812                                    ;               ERROR 32        LINE CLOCK FAILURE
4813                                    ;               ERROR 40        TRANSMIT INTERRUPT WHILE DISABLED
4814                                    ;               ERROR 35        UNDERRUN INTERRUPT
4815                                    ;               ERROR 43        UNEXPECTED UNDERRUN INTERRUPT AFTER ACK
4816                                    ;               ERROR 45        XMT/RCV PROCESSING FAILS
4817                                    ;               ERROR 46        UNEXPECTED STATUS RECEIVED
4818                                    ;               ERROR 47        UNEXPECTED DATA RECEIVED
4819                                    ;               ERROR 50        RECEIVE INTERRUPT WHILE DISABLED
4820                                    ;               ERROR 51        INTERRUPT AT ILLEGAL VECTOR
4821                                    ;--
4822 041210                            BGNTST
4823 041210  004737  034626                    CALL    STRSB0                  ;CALL START SUBROUTINE 0
4824 041214  004737  034646            1$:     CALL    STRSB1                  ;CALL START SUBROUTINE 1
4825 041220                                    TESTNB  30                      ;SELECT TEST30 IN FW
```

```
 4826 041226  004737  034710                    CALL    T18B00          ;CHECK ERROR
 4827 041232  022737  000007  002362            CMP     #7,RATE         ;NEXT LINE RATE REQUESTED ?
 4828 041240  001004                            BNE     2$              ;IF NOT
 4829 041242  005721                            TST     (R1)+           ;IF YES, SET RATE
 4830 041244  022701  000016                    CMP     #16,R1          ;ANY-MORE?
 4831 041250  001361                            BNE     1$              ;IF YES, LOOP
 4832 041252  000240                    2$:     NOP
 4833 041254                            ENDTST
 4834
 4835
```

```
                        BADHEAD
                        ;++
                        ;
                        ;
                        ;
                        ;                       ** TEST37 **
                        ;
                        ;
                        ;
4838                    ; LONG RAM TEST
4839                    ;
4840                    ; DESCRIPTION:  THE HOST ASKS THE DCT11 FOR TESTING THE RAM
4841                    ;               BY ITSELF. PERIODICALLY THE HOST CHECKS THE
4842                    ;               REGISTER SEL0 FOR ANY DCT11 RESPONSE:
4843                    ;                       SEL0 = 0        IF TEST SUCCEEDED
4844                    ;                       SEL0 = 100      IF ANY ERROR
4845                    ;               IN THE LATTER CASE, THE FOLLOWING REGISTERS
4846                    ;               CONTAINS THE ERROR REPORTS AS FOLLOW:
4847                    ;
4848                    ;               - IF THE DCT11 CAN'T CLEAR ALL THE RAM
4849                    ;                       SEL2 = 2
4850                    ;                       SEL4 = FIRST BAD WORD ADDRESS
4851                    ;                       SEL6 = FIRST BAD WORD
4852                    ;
4853                    ;               - IF DATA COMPARE ERROR AT LOCATION TESTED
4854                    ;                       SEL2 = 4
4855                    ;                       SEL4 = RAM ADDRESS
4856                    ;                       SEL6 = BAD WORD
4857                    ;                       SEL10= PATTERN
4858                    ;
4859                    ;               - IF WRITTEN PATTERN IN ONE LOCATION PRODU-
4860                    ;                 CES FALSE DATA IN OTHER LOCATIONS
4861                    ;                       SEL2 = 6
4862                    ;                       SEL4 = FIRST BAD WORD ADDRESS
4863                    ;                       SEL6 = FIRST BAD WORD
4864                    ;                       SEL10= PATTERN
4865                    ;                       SEL12= LOCATION UNDER TEST
4866                    ;
4867                    ; MAINT MODE:   2
4868                    ;
4869                    ; KMV ROUTINE:  05
4870                    ;
4871                    ; REPORTS:      ERROR 2         MASTER CLEAR FAILS TO RESET
4872                    ;               ERROR 11        NO ANSWER FROM THE KMV11
4873                    ;               ERROR 14        ALL THE RAM CAN'T BE CLEARED
4874                    ;               ERROR 15        DATA COMPARE ERROR ON RAM AT
4875                    ;                               LOCATION UNDER TEST
4876                    ;               ERROR 16        DATA COMPARE ERROR ON RAM AT
4877                    ;                               LOCATION NOT UNDER TEST
4878                    ;
4879                    ;--
4880
4881
4882 041256            BGNTST
4883 041256  005737  002406         TST     RMTST                   ; TEST TO BE DONE ?
```

```
4884 041262  001002                        BNE     20$                     ; YES CONTINUE
4885 041264                                EXIT    TST
4886
4887 041270                          20$:
4888 041270  013705  002474                MOV     KMVCSR,R5               ;GET CSR ADDRESS
4889 041274                                MODE    2                       ;SELECT MAINTENANCE MODE 1
4890 041300  103002                        BCC     1000$
4891 041302                                ESCAPE  TST
4892 041306                          1000$:
4893 041306                                TESTNB  5                       ;SELECT TEST 5 IN FW
4894 041314                                PRINTB  #MTST37                 ;PRINT A MESSAGE TO PREVENT USER OF
4895                                                                        ;THE TEST DURATION (ABOUT 1H15)
4896
4897 041334                                WAIT    1,65535.,10.            ;WAIT AND CHECK SELO
4898 041344  000474                        BR      6$                      ;SELO IS 0, CORRECT
4899 041346  000406                        BR      2$                      ;IF ANY ERROR
4900
4901 041350                                ERRHRD  11,EM0011               ;IF NO RESPONSE
4902 041360                                ESCAPE TST
4903
4904 041364                          2$:   DSPACH  3
4905 041372  000404                        BR      3$                      ;FOR ERROR 14
4906 041374  000417                        BR      4$                      ;FOR ERROR 15
4907 041376  000435                        BR      5$                      ;FOR ERROR 16
4908
4909 041400                                ESCAPE  TST
4910
4911 041404  016537  000004  002414  3$:   MOV     SEL4(R5),BADLOC         ;GET BAD RAM ADDRESS
4912 041412  016537  000006  002352        MOV     SEL6(R5),BAD            ;GET BAD VALUE
4913 041420                                ERRHRD  14,EM0013,PRRAM1        ;REPORT
4914 041430                                ESCAPE  TST
4915
4916 041434  016537  000004  002416  4$:   MOV     SEL4(R5),TSTLOC         ;GET SELECTED RAM ADDRESS
4917 041442  016537  000006  002412        MOV     SEL6(R5),BDDAT          ;GET BAD VALUE
4918 041450  016537  000010  002360        MOV     SEL10(R5),DATA          ;GET PATTERN
4919 041456                                ERRHRD  15,EM0014,PRRAM2        ;REPORT
4920 041466                                ESCAPE  TST
4921
4922 041472  016537  000004  002414  5$:   MOV     SEL4(R5),BADLOC         ;GET BAD RAM ADDRESS
4923 041500  016537  000006  002352        MOV     SEL6(R5),BAD            ;GET BAD VALUE
4924 041506  016537  000010  002360        MOV     SEL10(R5),DATA          ;GET PATTERN
4925 041514  016537  000012  002416        MOV     SEL12(R5),TSTLOC        ;GET SELECTED RAM ADDRESS
4926 041522                                ERRHRD  16,EM0015,PRRAM0        ;REPORT
4927 041532                                ESCAPE  TST
4928
4929 041536  000240                  6$:   NOP
4930 041540                          ENDTST
4931                                 .EVEN
```

```
                        .SBTTL  $SIZE   ROUTINE TO SIZE MEMORY


                        ;*************************************************************************
                        ;* NOTE:        THIS ROUTINE MUST RESIDE WITHIN THE
                        ;*              FIRST 24K OF MEMORY.
                        ;*
                        ;*
                        ;*
                        ;*      CALL :  JSR     PC,$SIZE
                        ;*$KT11 IS THE MEMORY MANAGEMENT KEY
                        ;*BIT07 = 0 DON'T USE MEMORY MANAGEMENT
                        ;*      MUST BE SETUP BEFORE THE CALL
                        ;*BIT15 = 0 DON'T HAVE MEMORY MANAGEMENT OPTION
                        ;*      DETERMINED BY ROUTINE
                        ;*$LSTAD WILL CONTAIN THE LAST AVAILABLE MEMORY LOCATION (BITS 15-0)
                        ;*$LSTBK WILL CONTAIN THE LAST AVAILABLE MEMORY LOCATION (BITS 21-16)
                        ;*************************************************************************


                        ;       HOW TO USE THIS SUBROUTINE WITH A D.M.A. INTERFACE ?


                        ;       1) CLR  $KT11   TO DISABLE MEMORY MANAGEMENT
                        ;       2) JSR  PC,$SIZE
                        ;               $LSTAD IS THE HIGHEST ADDRESS ACCESSIBLE BY SOFTWARE
                        ;               $LSTBK=0
                        ;       3) TEST D.M.A. AND DATA PATTERN IN THE MEMORY BETWEEN
                        ;               ADDRESS 0 AND THAT HIGHEST ADDRESS
                        ;       4) IF $LSTAD < 157776 , JUMP TO 9)
                        ;       5) MOV  @200,$KT11      TO ENABLE MEMORY MANAGEMENT
                        ;       6) JSR  PC,$SIZE
                        ;               $LSTAD,$LSTBK IS THE HIGHEST MEMORY ADDRESS
                        ;       7) IF $LSTAD,$LSTBK > 157776,3 , FORCE $LSTAD=157776
                        ;               AND $LSTBK=3 (124 K FOR A 18 BITS INTERFACE)
                        ;       8) TEST THAT D.M.A. IS POSSIBLE UP TO THAT ADDRESS
                        ;       9) TEST THAT D.M.A. GENERATES NXM BETWEEN THAT ADDRESS
                        ;               AND ADDRESS 760000 (ALWAYS UNUSED ON UNIBUS).
```

```
                000004                          ERRVEC=4
                172340                          KIPAR0=172340
                172356                          KIPAR7=172356
                177572                          SR0=177572
                172516                          SR3=172516
                                                
                                                
                                                
                                                
                                                
041542  010046                  $SIZE:  MOV     R0,-(SP)        ;;SAVE R0 ON THE STACK
041544  010146                          MOV     R1,-(SP)        ;;SAVE R1 ON THE STACK
041546  010246                          MOV     R2,-(SP)        ;;SAVE R2 ON THE STACK
041550  010346                          MOV     R3,-(SP)        ;;SAVE R3 ON THE STACK
041552  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE PRESENT ERROR VECTOR PS & PC
041556  013746  000006                  MOV     @#ERRVEC+2,-(SP)
041562  010600                          MOV     SP,R0           ;;SAVE THE STACK POINTER
                                ;;SET THE ERRVEC PS TO THE PRESENT PS
041564  106737  000006                  MFPS    @#ERRVEC+2
041570  012701  003776                  MOV     #3776,R1        ;;SETUP ADDRESS
041574  105727                          TSTB    (PC)+           ;;USE MEMORY MANAGEMENT?
041576  000200                  $KT11:  .WORD   200             ;;SET TO USE MEMORY MANAGEMENT
041600  100062                          BPL     $CORE           ;;BR IF NO
041602  012737  041740  000004          MOV     #$KTNEX,@#ERRVEC ;;SET FOR TIMEOUT
041610  005737  177572                  TST     @#SR0           ;;KT11 ARE YOU THERE?
041614  052737  100000  041576          BIS     #100000,$KT11   ;;YES--SET KT11 KEY
041622  005046                          CLR     -(SP)           ;;INITIALIZE FOR "PAR" LOADING
041624  012702  172340                  MOV     #KIPAR0,R2      ;;ADDRESS OF FIRST "PAR"
041630  012703  000010                  MOV     #^D8,R3         ;;LOAD EIGHT "PAR.'S" AND EIGHT "PDR.'S"
041634  012762  077406  177740  1$:     MOV     #77406,-40(R2)  ;;PDR = 4K, UP, READ/WRITE
041642  011622                          MOV     (SP),(R2)+      ;;LOAD "PAR"
041644  062716  000200                  ADD     #200,(SP)       ;;UPDATE FOR NEXT "PAR"
041650  077307                          SOB     R3,1$           ;;LOOP UNTIL ALL EIGHT ARE LOADED
041652  012742  177600                  MOV     #177600,-(R2)   ;;SETUP KIPAR7 FOR I/O
041656  005042                          CLR     -(R2)           ;;SETUP KIPAR6 FOR TESTING
041660  012737  041676  000004          MOV     #2$,@#ERRVEC    ;;CATCH TIMEOUT IF NO SR3
041666  012737  000020  172516          MOV     #20,@#SR3       ;;ENABLE 22 BIT MODE
041674  000401                          BR      3$              ;;THIS PDP-11 HAS A SR3 REGISTER
041676  022626                  2$:     CMP     (SP)+,(SP)+     ;;CLEAN OFF THE STACK--NO SR3
041700  005237  177572          3$:     INC     @#SR0           ;;TURN ON MEMORY MANAGEMENT
041704  012737  041730  000004          MOV     #$KTOUT,@#ERRVEC ;;SET FOR TIME OUT
041712  005737  143776          4$:     TST     @#143776        ;;TRAP ON NON-EX-MEM
041716  062712  000040                  ADD     #40,(R2)        ;;MAKE A 1K STEP
041722  023712  172356                  CMP     @#KIPAR7,(R2)   ;;LAST ONE?
041726  101371                          BHI     4$              ;;NO--TRY IT
041730  011202                  $KTOUT: MOV     (R2),R2         ;;GET LAST BANK+1
041732  005037  177572                  CLR     @#SR0           ;;TURN OFF MEMORY MANAGEMENT
041736  000421                          BR      $SIZEX
041740  042737  100000  041576  $KTNEX: BIC     #100000,$KT11   ;;KT11 NON-EXISTENT
041746  012737  041776  000004  $CORE:  MOV     #$CROUT,@#ERRVEC ;;SET FOR TIMEOUT
041754  005002                          CLR     R2              ;;SET UP BANK
041756  062701  004000          1$:     ADD     #4000,R1        ;;INCREMENT BY 1K
041762  062702  000040                  ADD     #40,R2          ;;1K STEP
041766  005711                          TST     (R1)            ;;TRAP ON TIME OUT
041770  022701  177776                  CMP     #177776,R1      ;;LAST ONE
041774  001370                          BNE     1$              ;;NO--TRY AGAIN
041776  162701  004000  $CROUT: SUB     #4000,R1
```

```
 5039 042002  005737  002364          $SIZEX: TST     NXMTST          ;NXM FLAG SET ?
 5040 042006  001002                          BNE     2$              ;YES KEEP NXM ADDRESS
 5041 042010  162702  000040                  SUB     #40,R2          ;;DROP BACK
 5042 042014  010006                  2$:     MOV     R0,SP           ;;RESTORE THE STACK
 5043 042016  012637  000006                  MOV     (SP)+,@#ERRVEC+2 ;;RESTORE ERROR VECTOR
 5044 042022  012637  000004                  MOV     (SP)+,@#ERRVEC
 5045 042026  010137  042066                  MOV     R1,$LSTAD       ;;LAST ADDRESS (BITS 10-0)
 5046 042032  000241                          CLC
 5047 042034  006002                          ROR     R2              ;;ROTATE BANK
 5048 042036  006002                          ROR     R2
 5049 042040  150237  042067                  BISB    R2,$LSTAD+1     ;;LAST ADDRESS (BITS 15-11)
 5050 042044  105002                          CLRB    R2              ;;CLEAR BITS 15-11
 5051 042046  000302                          SWAB    R2              ;;SWAP R2
 5052 042050  010237  042070                  MOV     R2,$LSTBK       ;;LAST ADDRESS (BITS 21-16)
 5053 042054  012603                          MOV     (SP)+,R3        ;;RESTORE R3
 5054 042056  012602                          MOV     (SP)+,R2        ;;RESTORE R2
 5055 042060  012601                          MOV     (SP)+,R1        ;;RESTORE R1
 5056 042062  012600                          MOV     (SP)+,R0        ;;RESTORE R0
 5057 042064  000207                          RTS     PC
 5058
 5059
 5060 042066  000000          $LSTAD: .WORD   0               ;;CONTAINS THE LAST ADDRESS (BITS 15-0)
 5061 042070  000000          $LSTBK: .WORD   0               ;;CONTAINS THE LAST ADDRESS (BITS 21-16)
```

```
                                .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.
                                ;////////////////////////////////////////////////////////////////////////////////

042072                                  BGNHRD

042074                                  GPRMA   ADDRES,0,0,160000,177776,YES
042104                                  GPRMA   VECTOR,2,0,0,674,YES
                                ;       GPRMD   PRIRTY,4,0,7000,4,7,YES
042114                                  GPRML   LOOPBK,6,-1,YES
042122                                  GPRMD   LINRAT,10,0,7,0,7,YES
042134                                  GPRML   ST37,12,-1,YES                  ; /V01.04/
042142                                  ENDHRD


                                .IF NE HELP
                                ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                ;       INSERT MESSAGES THAT ARE USED ONLY
                                ;       DURING THE HARDWARE PARAMETER CODING SECTION.
                                ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                .ENDC


042142  104  105  126  ADDRES: .ASCIZ  /DEVICE CSR ADDRESS : /
042170  104  105  126  VECTOR: .ASCIZ  /DEVICE FIRST VECTOR ADDRESS : /
042227  104  105  126  PRIRTY: .ASCIZ  /DEVICE PRIORITY LEVEL : /
042260  114  117  117  LOOPBK: .ASCIZ  /LOOPBACK CONNECTOR USED ? : /
042315  104  105  126  LINRAT: .ASCIZ  /DEVICE LINE CLOCK RATE SELECTED : /
042360  104  117  040  ST37:   .ASCIZ  /DO YOU WANT TO RUN LONG RAM TEST ?  /
                                .EVEN



                                .IF NE HELP
                                ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                ;       INSERT SOFTWARE PARAMETER INTERPRETIVE CODING HERE.  THIS CODE
                                ;       IS USED BY THE SUPERVISOR TO INTERROGATE THE OPERATOR FOR
                                ;       SOFTWARE INFORMATION WHICH WILL BE PLACED IN THE SOFTWARE
                                ;       TABLE.  THIS SECTION IS OPTIONAL.
                                ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                .ENDC
```

```
5120                                    .IF NE HELP
5121                                    ;*****************************************************************************
5122                                    ;       INSERT MESSAGES THAT ARE USED ONLY
5123                                    ;       DURING THE SOFTWARE PARAMETER CODING SECTION.
5124                                    ;*****************************************************************************
5125                                    .ENDC
5126
5127
```

```
5129
5130 042426                            $PATCH::
5131
5132                                   .IF NE HELP
5133                                   ;*************************************************************************************
5134                                   ;       THIS IS A PATCH AREA THAT SHOULD BE INCLUDED IN ALL DIAGNOSTICS.
5135                                   ;       ADJUST THE SIZE TO FIT YOUR OWN PREFERENCES.
5136                                   ;*************************************************************************************
5137                                   .ENDC
5138
5139 042426                                    .BLKB   200
5140 042626                                    LASTAD
     042632                            L$LAST::
5141 042632                                    ENDMOD
5142
5143
```

```
5145
5146
5147                                .IF NE HELP
5148                                ;*****************************************************************************************
5149                                ;       HARDCODED P-TABLES MAY BE PLACED HERE BY USING THE SETUP MACROS.
5150                                ;       THIS SECTION IS OPTIONAL AND SHOULD BE REMOVED IF IT IS NOT BEING
5151                                ;       USED.  CHANGE THE POINTER MACRO ARGUMENT TO REFLECT THE REMOVAL.
5152                                ;
5153                                ;       THE P-TABLES ARE DELIMITED BY THE "BGNSETUP" AND "ENDSETUP" MACROS.
5154                                ;       THE "BGNSETUP" MACRO HAS ONE ARGUMENT WHICH IS THE NUMBER OF
5155                                ;       P-TABLE ENTRIES.   EACH ENTRY IS DELIMITED BY THE "BGNPTAB" AND
5156                                ;       "ENDPTAB" MACROS.  NEITHER OF THESE MACROS REQUIRE AN ARGUMENT.
5157                                ;*****************************************************************************************
5158                                .ENDC
5159                                ;       BGNSETUP        1
5160                                ;       BGNPTAB
5161                                ;       .WORD   177000
5162                                ;       .WORD   300
5163                                ;       .WORD   4000
5164                                ;       .WORD   1
5165                                ;       .WORD   3
5166                                ;       .WORD   0               ; /V01.04/
5167                                ;       ENDPTAB
5168                                ;       ENDSETUP
5169
5170        000001                  .END
```

SYMBOL TABLE

```
A     = 000377          C$CVEC= 000036          EF.NEW= 000035 G        ERR   = 000400          GOOD    002350
ADDR    002520          C$DCLN= 000044          EF.PWR= 000034 G        ERRBLK  002332 G        G$CNTO= 000200
ADDRES  042142          C$DODU= 000051          EF.RES= 000037 G        ERRCNT  002524          G$DELM= 000372
ADR   = 000020 G        C$DRPT= 000024          EF.STA= 000040 G        ERRMSG  002330 G        G$DISP= 000003
ASSEMB= 000010          C$DU  = 000053          EM0000  011166          ERRNBR  002326 G        G$EXCP= 000400
BAD     002352          C$EDIT= 000003          EM0001  011232          ERRTYP  002324 G        G$HILI= 000002
BADLOC  002414          C$ERDF= 000055          EM0002  011317          ERRVEC= 000004          G$LOLI= 000001
BDDAT   002412          C$ERHR= 000056          EM0003  011364          EVB10   026610          G$NO  = 000000
BDLVL   002420          C$ERRO= 000060          EM0004  011452          EVB2    026310          G$OFFS= 000400
BIT0  = 000001 G        C$ERSF= 000054          EM0005  011524          EVB3    026370          G$OFSI= 000376
BIT00 = 000001 G        C$ERSO= 000057          EM0006  011614          EVB4    026450          G$PRMA= 000001
BIT01 = 000002 G        C$ESCA= 000010          EM0007  011662          EVB5    026530          G$PRMD= 000002
BIT02 = 000004 G        C$ESEG= 000005          EM0010  011720          EVL   = 000004 G        G$PRML= 000000
BIT03 = 000010 G        C$ESUB= 000003          EM0011  011761          EVR10   024626          G$RADA= 000140
BIT04 = 000020 G        C$ETST= 000001          EM0012  012006          EVR2    024546          G$RADB= 000000
BIT05 = 000040 G        C$EXIT= 000032          EM0013  012056          EVTT2   023472          G$RADD= 000040
BIT06 = 000100 G        C$GETB= 000026          EM0014  012116          EVT0    021270          G$RADL= 000120
BIT07 = 000200 G        C$GETW= 000027          EM0015  012173          EVT1    021350          G$RADO= 000020
BIT08 = 000400 G        C$GMAN= 000043          EM0016  012261          EVT10   022070          G$XFER= 000004
BIT09 = 001000 G        C$GPHR= 000042          EM0017  012345          EVT11   022150          G$YES = 000010
BIT1  = 000002 G        C$GPLO= 000030          EM0020  012426          EVT12   022230          HELP  = 000000
BIT10 = 002000 G        C$GPRI= 000040          EM0021  012464          EVT13   022310          HOE   = 100000 G
BIT11 = 004000 G        C$INIT= 000011          EM0022  012542          EVT15   022370          IBE   = 010000 G
BIT12 = 010000 G        C$INLP= 000020          EM0023  012622          EVT2    021430          IDU   = 000040 G
BIT13 = 020000 G        C$MANI= 000050          EM0024  012717          EVT3    021510          IER   = 020000 G
BIT14 = 040000 G        C$MEM = 000031          EM0025  012773          EVT4    021570          INDAT   027503
BIT15 = 100000 G        C$MSG = 000023          EM0026  013061          EVT7    021650          INIFLG  002510
BIT2  = 000004 G        C$OPEN= 000034          EM0027  013127          EVT8    021730          INREG   027134 G
BIT3  = 000010 G        C$PNTB= 000014          EM0030  013223          EVT9    022010          INRG1   027244
BIT4  = 000020 G        C$PNTF= 000017          EM0031  013271          E$END = 002100          INRG2   027276
BIT5  = 000040 G        C$PNTS= 000016          EM0032  013334          E$LOAD= 000035          INRG3   027331
BIT6  = 000100 G        C$PNTX= 000015          EM0033  013426          FLGP11  002342          INRG4   027365
BIT7  = 000200 G        C$QIO = 000377          EM0034  013474          FTIME   002340          INRG5   027416
BIT8  = 000400 G        C$RDBU= 000007          EM0037  013527          F$AU  = 000015          INRG6   027450
BIT9  = 001000 G        C$REFG= 000047          EM0040  013620          F$AUTO= 000020          INTFLG  002356
BOE   = 000400 G        C$RESE= 000033          EM0041  013700          F$BGN = 000040          ISR   = 000100 G
BSELIT  033370          C$REVI= 000003          EM0042  013745          F$CLEA= 000007          IXE   = 004000 G
BUFLEN  002470          C$RFLA= 000021          EM0043  014045          F$DU  = 000016          I$AU  = 000041
BUF01   002430          C$RPT = 000025          EM0044  014123          F$END = 000041          I$AUTO= 000041
CHKMAX  007604          C$SEFG= 000046          EM0045  014211          F$HARD= 000004          I$CLN = 000041
CLEAR   011052          C$SPRI= 000041          EM0046  014244          F$HW  = 000013          I$DU  = 000041
CLRKMV  011116          C$SVEC= 000037          EM0047  014275          F$INIT= 000006          I$HRD = 000041
COMFLG  020446          C$TPRI= 000013          EM0050  014356          F$JMP = 000050          I$INIT= 000041
COMPAR  010356          DATA    002360          EM0051  014440          F$MOD = 000000          I$MOD = 000041
COMPRB  010444          DFPTBL  002246 G        EM0052  007032          F$MSG = 000011          I$MSG = 000041
COPY    011074          DIAGMC= 000000          EM0054  014505          F$PROT= 000021          I$PROT= 000040
CSRLEN= 000020          DMAINS  035470          EM0055  014555          F$PWR = 000017          I$PTAB= 000041
C$AU  = 000052          DMAOSB  036610          EM0056  014632          F$RPT = 000012          I$PWR = 000041
C$AUTO= 000061          DMARG1  024706          EM0057  014666          F$SEG = 000003          I$RPT = 000041
C$BRK = 000022          DMARG2  024761          EM0060  014735          F$SOFT= 000005          I$SEG = 000041
C$BSEG= 000004          DMAR1   025220          EM0061  015004          F$SRV = 000010          I$SETU= 000041
C$BSUB= 000002          DMAR2   025305          EM0062  015053          F$SUB = 000002          I$SRV = 000041
C$CEFG= 000045          DMASUB  037350          EM0063  015122          F$SW  = 000014          I$SUB = 000041
C$CLCK= 000062          DMATG1  023552          EM0064  015155          F$TEST= 000001          I$TST = 000041
C$CLEA= 000012          DMATG2  023625          EM0065  015247          GDDAT   002410          J$JMP = 000167
C$CLOS= 000035          DROPD   030232          ENDINI  027776          GDLVL   002422          KIPAR0= 172340
C$CLP1= 000006          EF.CON= 000036 G        ENDT33  040506          GDVEC   002424          KIPAR7= 172356
```

SYMBOL TABLE

```
KMVCSR  002474          L$SPCP  002020 G        L10063  032516          MRAM2   017174          PRIRAT  020052
KMVLVL  002502          L$SPTP  002024 G        L10064  032702          MRATE   020173          PRIRTY  042227
KMVV00  002476          L$STA   002030 G        L10065  033020          MREV    016570          PRITRX  020224 G
KMVV04  002500          L$SW    002504          L10066  033126          MSCHEK  016470          PRI00 = 000000 G
LCLOOP  002366          L$TEST  002114 G        L10067  033056          MSEL    016104          PRI01 = 000040 G
LINRAT  042315          L$TIML  002014 G        L10070  033116          MSTACK  016362          PRI02 = 000100 G
LOCK    002506          L$UNIT  002012 G        L10071  033606          MTMOTX  020270          PRI03 = 000140 G
LOE   = 040000 G        L10001  002262          L10072  033334          MTST37  007116          PRI04 = 000200 G
LOGDEV  002334          L10002  015402          L10073  033362          MVEC    017520          PRI05 = 000240 G
LOKFLG  002512          L10003  015540          L10074  034014          NERRS   007676          PRI06 = 000300 G
LOOPBK  042260          L10004  015576          L10075  034326          NEXT    027660          PRI07 = 000340 G
LOT   = 000010 G        L10005  015756          L10076  034374          NUM0    002402          PRLVL   017310 G
L$ACP   002110 G        L10006  016102          L10077  034442          NUM1    002404          PRMDM   026670 G
L$APT   002036 G        L10007  016326          L10100  034510          NXMHIG  002530          PRPER   017746 G
L$AU    030266 G        L10010  016360          L10101  034556          NXMLOW  002526          PRRAM0  016650 G
L$AUT   002070 G        L10011  016566          L10102  034624          NXMTST  002364          PRRAM1  016736 G
L$AUTO  030106 G        L10012  016734          L10103  035260          O$APTS= 000000          PRRAM2  016774 G
L$CCP   002106 G        L10013  016772          L10104  035360          O$AU  = 000001          PRRATE  020040 G
L$CLEA  030200 G        L10014  017034          L10105  035466          O$BGNR= 000000          PRREG   015600 G
L$CO    002032 G        L10015  017350          L10106  036256          O$BGNS= 000000          PRREV   016526 G
L$DEPO  002011 G        L10016  017516          L10107  036500          O$DU  = 000001          PRSEL   015500 G
L$DESC  002262 G        L10017  017616          L10110  036606          O$ERRT= 000000          PRSTAK  016266 G
L$DESP  002076 G        L10020  017704          L10111  037150          O$GNSW= 000001          PRVEC   017466 G
L$DEVP  002060 G        L10021  017776          L10112  037222          O$POIN= 000001          PSTACK  002336
L$DISP  002132 G        L10022  020050          L10113  037274          O$SETU= 000001          QBUSIT  033130
L$DLY   002116 G        L10023  020266          L10114  037346          PADFLT  015346 G        QV.FLG  002513
L$DTP   002040 G        L10024  020360          L10115  040510          PARTNB  006764          RATE    002362
L$DTYP  002034 G        L10025  022460          L10116  041070          PATLGH= 000112          RATETB  020106
L$DU    030204 G        L10026  022656          L10117  040610          PATTRN  007204          RATE0   020124
L$DUT   002072 G        L10027  022674          L10120  040640          PDMAB1  025572 G        RATE1   020132
L$DVTY  007576 G        L10030  022712          L10121  040670          PDMAB2  025610 G        RATE2   020140
L$EF    002052 G        L10031  024052          L10122  041206          PDMARG  023736          RATE3   020147
L$ENVI  002044 G        L10032  024070          L10123  041254          PDMART  025560 G        RATE4   020154
L$ERRT  002324 G        L10033  024106          L10124  041540          PDMAR1  024042 G        RATE5   020161
L$ETP   002102 G        L10034  025216          L10125  042142          PDMAR2  024054 G        RATE6   020166
L$EXP1  002046 G        L10035  025570          MAIERR  010606          PDMAR3  024072 G        RATIO   002426
L$EXP4  002064 G        L10036  025606          MAINM2  010634          PDMATG  022542          RATIOT  007356
L$EXP5  002066 G        L10037  025624          MAINM3  010522          PDMAT1  022646 G        RCVBFH  040040
L$HARD  042074 G        L10040  026750          MAINT1= 001000          PDMAT2  022660 G        RCVBFL  040042
L$HIME  002120 G        L10041  027224          MAINT2= 004000          PDMAT3  022676 G        RCVBUF  004626
L$HPCP  002016 G        L10042  027776          MAXERR  002522          PDMRTG  025372          RCVLNG= 001000
L$HPTP  002022 G        L10043  030176          MBAD    016161          PEVTBG  025626          READ  = 002000
L$HW    002246 G        L10044  030202          MCLR  = 040000          PEVTRG  024110          REGIN   027226
L$ICP   002104 G        L10045  030264          MCSR    017620          PEVTTG  022714          REGSPC  002532
L$INIT  027556 G        L10046  030266          MDAT    021126          PGCSR   017654 G        REGTAB  007316
L$LADP  002026 G        L10047  030406          MEVT    021227          PNT   = 001000 G        REGTB   027230
L$LAST  042632 G        L10050  030516          MGCSR   017706          PNTBL   006622          REG1    002370
L$LOAD  002100 G        L10051  030524          MLVL    017352          PRALL   015760 G        REG2    002372
L$LUN   002074 G        L10052  030700          MMOD0   026752          PRBAD   015542 G        REG3    002374
L$MREV  002050 G        L10053  031046          MMOD1   027026          PRCHEK  016330 G        REG4    002376
L$NAME  002000 G        L10054  031266          MNULL   021214          PRCSR   017566 G        REG5    002400
L$PRIO  002042 G        L10055  031170          MPAT    016212          PRDAT   020324 G        REVCOD  002472
L$PROT  002122 G        L10056  031260          MPER    020000          PRDATG  020362          RMTST   002406
L$PRT   002112 G        L10057  031370          MPNUMH  006723          PRDMAR  025072 G        RUN   = 100000
L$REPP  002062 G        L10060  031570          MPNUML  006662          PREVT   022450 G        RUNNIN  030000
L$REV   002010 G        L10061  031756          MRAM0   017036          PREVTG  022462          RXLNG1= 000017
L$SPC   002056 G        L10062  032222          MRAM1   017122          PRI   = 002000 G        RXLNG2= 000010
```

SYMBOL TABLE

```
RXLNG3= 000006          SVCTST= 177777          T$PTNU= 000000          T15.2   033336          T6      031050 G
RXLNG4= 001774          S$LSYM= 010000          T$SAVL= 177777          T16     033610 G        T6.1    031050
SAVE4   002344          TDMALG= 001774          T$SEGL= 177777          T17     034016 G        T6.2    031172
SAVE6   002346          TFM36   015404          T$SIZE= ****** GX       T18     034330 G        T7      031270 G
SEL0  = 000000          TIM     011150          T$SUBN= 000000          T18B00  034710          T8      031372 G
SEL10 = 000010          TRXBFH= 000000          T$TAGL= 177777          T19     034476 G        T9      031572 G
SEL12 = 000012          TRXBFL= 107700          T$TAGN= 010126          T2      030410 G        UAM   = 000200 G
SEL14 = 000014          TRXBUF  002632          T$TEMP= 000000          T20     034444 G        UNIT    002516
SEL16 = 000016          TRXLNG= 001000          T$TEST= 000045          T21     034512 G        UUT     002514
SEL2  = 000002          TSTLOC  002416          T$TSTM= 177777          T22     034560 G        VECTOR  042170
SEL20 = 000020          TSTNB   010772          T$TSTS= 000001          T23     035164 G        VKMHAO  002000 G
SEL22 = 000022          TSTNUM= 010000          T$$AU = 010046          T23A00  040676          WAIT0   010070
SEL24 = 000024          TSTSL2  007746          T$$AUT= 010043          T23A01  041070          WAIT1   010154
SEL26 = 000026          TXLNG1= 000013          T$$CLE= 010044          T24     035262 G        WAIT2   010304
SEL3  = 000003          TXLNG2= 000012          T$$DU = 010045          T25     035362 G        WRITE = 020000
SEL30 = 000030          TXLNG4= 001774          T$$HAR= 010125          T26     036040 G        WRKSPC  002572
SEL32 = 000032          T$ARGC= 000001          T$$HW = 010001          T27     036260 G        X$ALWA= 000000
SEL34 = 000034          T$CODE= 005130          T$$INI= 010042          T28     036502 G        X$FALS= 000040
SEL36 = 000036          T$ERRN= 000020          T$$MSG= 010041          T29     037100 G        X$OFFS= 000400
SEL4  = 000004          T$EXCP= 000000          T$$PRO= 010000          T3      030520 G        X$TRUE= 000020
SEL6  = 000006          T$FLAG= 000040          T$$SUB= 010121          T30     037152 G        $CORE   041746
SETUP   027652          T$FREE= ****** GX       T$$TES= 010124          T31     037224 G        $CROUT  041776
SPADDR  002354          T$GMAN= 000000          T1      030270 G        T32     037276 G        $KTNEX  041740
SR0   = 177572          T$HILI= 000007          T10     031760 G        T33     040044 G        $KTOUT  041730
SR3   = 172516          T$LAST= 000001          T11     032224 G        T34     040512 G        $KT11   041576
SSTACK  007576          T$LOLI= 000000          T12     032520 G        T34.1   040562          $LSTAD  042066
STRSB0  034626          T$LSYM= 010000          T13     032704 G        T34.2   040612          $LSTBK  042070
STRSB1  034646          T$LTNO= 000045          T14     033022 G        T34.3   040642          $LSTIN= 177777
ST37    042360          T$NEST= 177777          T14A00  033304          T35     041072 G        $LSTTA= 177777
SVCGBL= 000000          T$NS0 = 000000          T14.1   033022          T36     041210 G        $PATCH  042426 G
SVCINS= 177777          T$NS1 = 000004          T14.2   033060          T37     041256 G        $SIZE   041542
SVCSUB= 177777          T$NS2 = 000002          T15     033312 G        T4      030526 G        $SIZEX  042002
SVCTAG= 177777          T$PTHV= ****** GX       T15.1   033312          T5      030702 G

. ABS.  042632     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  30000 WORDS  ( 118 PAGES)
DYNAMIC MEMORY:  21924 WORDS  ( 84 PAGES)
ELAPSED TIME:  00:06:02
VKMHAO.BIN,VKMHAO.SEQ/-SP=LIBA/ML,VKMHAO.DOC,VKMHAO.P11
```