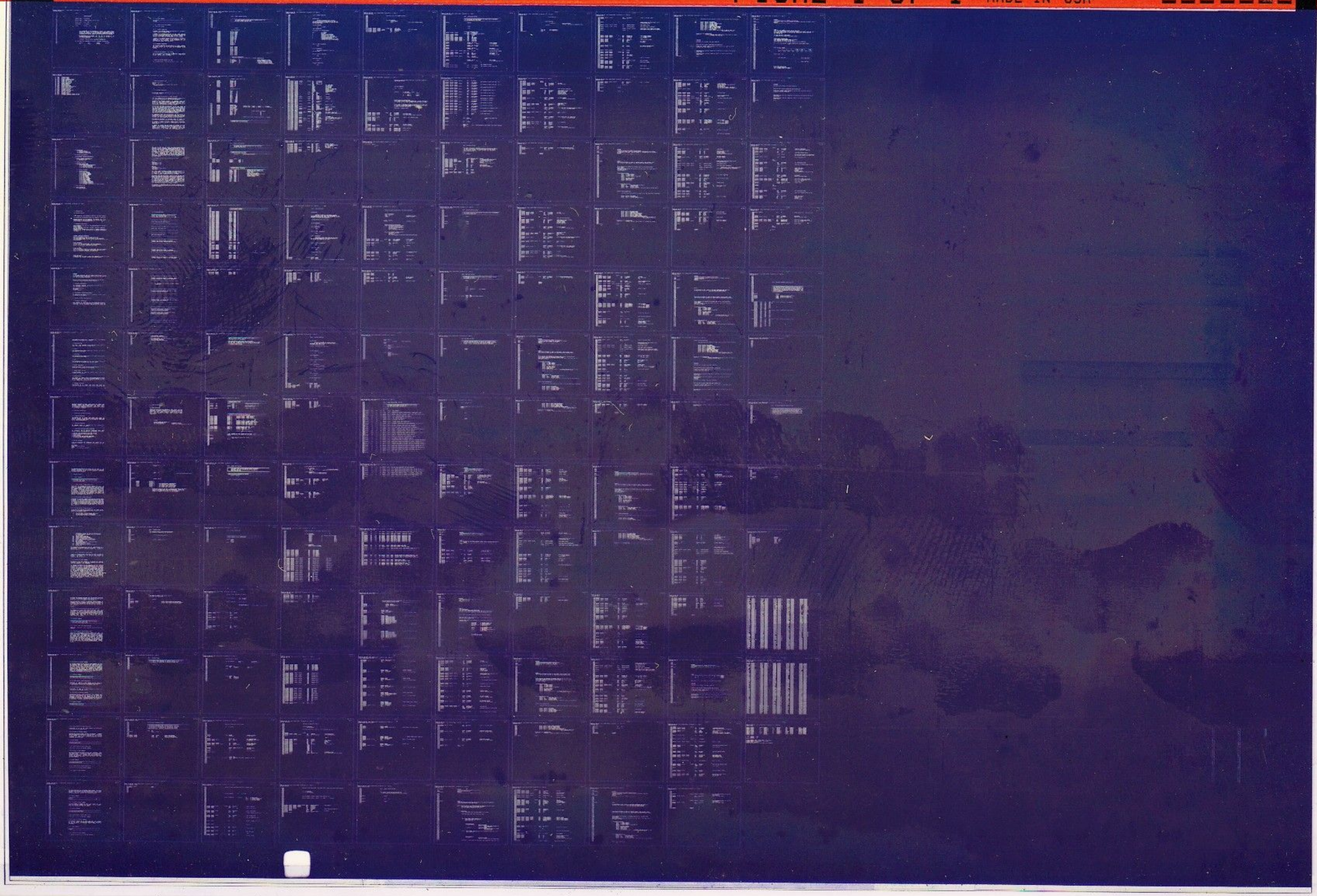
KMV11-B

KMV11-B
LINE CNT DIAG
CVKMEBO

AH-T378B-MC
FICHE 1 OF 1



```
                        .REM @

                                IDENTIFICATION
                                --------------

                        PRODUCT CODE:   AC-T377B-MC

                        PRODUCT NAME:   CVKMEB0 KMV11B LINE CNT DIAG

                        PRODUCT DATE:   APRIL 1983

                        MAINTAINER:     CSS ANNECY

                        AUTHOR:         MICHELET, GUY
```

TABLE OF CONTENTS

TABLE OF CONTENTS

```
                                1.0 INTRODUCTION

                                1.1 PROGRAM ABSTRACT


                                THIS  DIAGNOSTIC  WAS DESIGNED TO TEST OUT THE KMV11 MODULE

                                THE PROGRAM WAS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR.

                                THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM  WILL  ALLOW
                                MODIFICATION  OF  DEVICE PARAMETERS, SUCH AS QBUS ADDRESS,
                                VECTOR ADDRESS, AND PROCESSOR TYPE.


                                1.2 HARDWARE INTRODUCTION

                                THIS DIAGNOSTIC WILL TEST ALL THE HARDWARE PART OF THE KMV11 B
                                MODULE (M7501).
                                TO TEST COMPLETELY THIS PART ,EXTERNAL LOOP BACK CONNECTOR
                                MUST BE INSTALLED.
                                DIAGNOSTIC WILL AUTOMATICALLY DETECT IF LOOPBACK CONNECTOR IS
                                PLUGGED OR NOT (IF NOT ,EXTERNAL TEST IS DROPPED AND REPORTS
                                THE ERROR)



                                EXTERNAL LOOP BACK CONNECTOR:
                                -----------------------------
                                KMV11 B CAN OPERATE EITHER IN RS422 OR RS 423
                                FOR RS422 MODEM SIGNAL 103,104,114,AND 115 ARE SUPPORTED.
                                FOR RS 423 MODEM SIGNAL 103,104,105,107,108,106,109,113,114,115
                                ARE SUPPORTED.


                                RS422 LOOP BACK:
                                TO TEST  KMV11 B IN RS422 MODE ,RUN THIS DIAGNOSTIC
                                WITH THE ZIF LOOP BACK CONNECTOR 2P-E155A-00 PLUG ON THE ZIF
                                SOCKET (12-11591-35)AT THE END OF BC05 CABLE


                                RS423 LOOP BACK:
                                TO TEST COMPLETELY A KMV11 B IN RS423 MODE ,RUN THIS DIAGNOSTIC
                                WITH  ZIF LOOP BACK CONNECTOR  2P-E156A-00 PLUG ON THE ZIF
                                SOCKET (12-11591-35) AT THE END OF BC05 CABLE.



                                RS423 LOOP BACK:
                                DIAGNOSTIC WILL TEST KMv11 CLOCKS,LINE INTERUPTS,TX AND RX FUCTION
                                IN INTERNAL AND EXTERNAL LOOP BACK AND MODEM SIGNALS.
```

```
                                CAUTION:
                                *******
                                IF LOOP BACK CONNECTORS ARE NOT PLUGED IN BOTH CHANEL A AND B,
                                THE DIAGNOSTIC WILL AUTOMATICALY REPORT AN ERROR AND DROP THE
                                TEST FOR THE EXTERNAL LOOP BACK.


                                KMV11 B IS FULLY TESTED ONLY WHEN DIAGNOSTIC HAS BEEN RUN
                                SUCCESFULLY IN BOTH RS422 AND RS423 LOOP BACK.


                                2.0 HARDWARE REQUIREMENTS

                                THE  FOLLOWING  HARDWARE  IS REQUIRED TO RUN THE KMV11B
                                LINE CONTROLER STATIC TESTS:

                                PDP-11/03,23,23 PLUS
                                16K MEMORY
                                CONSOLE TERMINAL


                                3.0 PRELIMINARY PROGRAM REQUIREMENTS

                                THE PROCESSOR AND MEMORY SHOULD BE THOROUGHLY  TESTED  PRIOR
                                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 KMV11 STATIC TESTS IS ABOUT 190 SECONDS 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

THE NUMBER OF ERRORS WHICH HAVE OCCURRED ON EACH DEVICE UNDER TEST SINCE THE LAST START OR RESTART COMMAND IS KEPT IN AN ERROR LOG.   THIS LOG MAY BE PRINTED BY USING THE "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:

```
DRS LOADED
DIAG.  RUN-TIME SERVICES
VKMEB0
KMV11 B  LINE CONTROLER DIAGNOSTIC
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) CONTAIN-
                        ING THE ERROR
                    IER INHIBIT ERROR REPORTING
                    IBE INHIBIT BASIC ERROR REPORTS
                    IXE INHIBIT EXTENDED ERROR REPORTS
                    PRI DIRECT ALL MESSAGES TO A LINE PRINTER
                    PNT PRINT NUMBER OF TEST BEING EXECUTED
                    BOE BELL ON ERROR
                    UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL
                        INTERVENTION TESTS
                    ISR INHIBIT STATISTICAL REPORTS
                    IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
                    LOT LOOP ON TEST

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


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

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


                6.3.1.5 EFFECT OF START COMMAND

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

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

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

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

EXAMPLE:

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

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

6.3.2 RESTART COMMAND

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

6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

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

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

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

6.3.2.3 EFFECT OF RESTART COMMAND

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

6.3.3 CONTINUE COMMAND

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

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

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART. IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

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

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

6.3.3.3 EFFECT OF CONTINUE COMMAND

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

6.3.4 PROCEED COMMAND

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

```
530
531
532
533
534
535                         6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
536
537                         <FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED
538                         FLAGS RETAIN THEIR CURRENT VALUE.
539
540
541                         6.3.4.2 EFFECT OF PROCEED COMMAND
542
543                         PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE.  COMMAND
544                         MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR.  THE EFFECT
545                         OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION
546                         FOLLOWING THE ERROR CALL.  NEITHER HARDWARE NOR SOFTWARE
547                         PARAMETERS MAY BE ALTERED.
548
549
550                         6.3.5 ADD COMMAND
551
552                         ************************************************************
553                         ADD/UNITS:<UNIT-LIST>
554                         ************************************************************
555
556
557                         6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>
558
559                         <UNIT-LIST> IS AS IN THE RESTART COMMAND.
560
561
562                         6.3.5.2 EFFECT OF ADD COMMAND
563
564                         THE UNITS SPECIFIED ARE ADDED TO THE  TEST  SEQUENCE.  EACH
565                         UNIT  MUST  HAVE  A  P-TABLE  IN  MEMORY  DUE  TO AN EARLIER
566                         HARDWARE DIALOGUE.  THIS  COMMAND  MUST  BE  FOLLOWED  BY  A
567                         RESTART  OR  CONTINUE.  THE UNITS SWITCH MUST BE SPECIFIED.
568                         THE ADD COMMAND IS  MEANINGFUL  ONLY  FOR  UNITS  THAT  WERE
569                         PREVIOUSLY DROPPED.
570
571
572                         6.3.6 DROP COMMAND
573
574                         ************************************************  ****  ****
575                         DRO(P)/UNITS:<UNIT-LIST>
576                         ************************************************************
577
578
579                         6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
580
581                         <UNIT-LIST> IS AS IN THE RESTART COMMAND.
582
583
584                         6.3.6.2 EFFECT OF DROP COMMAND
```

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

6.3.7 PRINT COMMAND

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

6.3.7.1 EFFECT OF PRINT COMMAND

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

6.3.8 DISPLAY COMMAND

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

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

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

6.3.8.2 EFFECT OF DISPLAY COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.

6.3.9 FLAGS COMMAND

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

6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696

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 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.

2. MICRO-CPU CSR ADDRESS: (O) 177000?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SELO) RESIDE ON THE UNIBUS. THE ALLOWABLE RANGE IS 160000-177776 (OCTAL), AND THE DEFAULT IS 177000.

3. MICRO CPU VECTOR ADDRESS: (O) 300?

THE ALLOWABLE RANGE IS 300-770,AND DEFAULT VALUE IS 300

4. MICRO CPU PRIORITY LEVEL: (4) 7?

DEFFAULT VALYE IS 4
NOTE:
 M7501 MODULE MOUNTED WITH DC003 CHIPS CAN ONLY
  INTERUPT ON LEVEL 4

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY PART 2 OF THE STATIC LOGIC TESTS.

6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

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

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

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

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

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

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


                                ********************* TEST 1 *********************
                                *VERIFY THAT REFERENCED QBUS DEVICE REGISTERS
                                *DO NOT CAUSE TIME OUT TRAP
                                ***************************************************


                                ********************* TEST 2 *********************
                                *
                                *CHECK PROM REVISION COMPATIBILITY
                                *
                                ***************************************************


                                ********************* TEST 3 *********************
                                *
                                *KMV11 REAL TIME CLOCK TEST
                                *
                                ***************************************************


                                ********************* TEST 4 *********************
                                *
                                *BAUD RATE GENERATOR TEST
                                *
                                ***************************************************


                                ********************* TEST 5 *********************
                                *
                                *TRANSMIT AND RECEIVE FRAMES IN INTERNAL
                                *LOOPBACK MODE WITHOUT INTERRUPTIONS ON CHANEL A.
                                *
                                ***************************************************



                                ********************* TEST 6 *********************
                                *
                                *TRANSMIT AND RECEIVE FRAMES  IN INTERNAL
                                *LOOPBACK MODE WITHOUT INTERRUPTIONS ON CHANEL B.
                                *
                                ***************************************************


                                ********************* TEST 7 *********************
                                *
                                *TRANSMIT AND RECEIVE FRAMES AT DIFFERENT SPEEDS IN
                                *INTERNAL LOOPBACK ON CHANEL A  WITH INTERRUPTS.
```

```
                        *
                        ********************************************************



                        ********************** TEST 8 ********************
                        *
                        *TRANSMIT AND RECEIVE FRAMES AT DIFFERENT SPEEDS IN
                        *INTERNAL LOOPBACK ON CHANEL B WITH INTERRUPTS
                        *
                        ********************************************************


                        NOTE:  THE REMAINING TESTS REQUIRE EXTERNAL LOOPBACK
                                CONNECTORS.




                        ********************** TEST 9 ********************
                        *
                        *TRANSMIT FRAMES IN EXTERNAL LOOP BACK ON CHANEL A
                        *(LOOPBACK CONNECTOR INSTALLED)
                        *
                        ********************************************************



                        ********************** TEST 10 ********************
                        *
                        *TRANSMIT FRAMES  IN EXTERNAL LOOP BACK ON CHANEL B
                        *(LOOPBACK CONNECTOR INSTALLED)
                        *
                        ********************************************************



                        ********************** TEST 11 ********************
                        *
                        *TEST MODEM SIGNAL CCITT 107 AND CCITT 108 ON
                        *CHANEL A  WITH EXTERNAL LOOPBACK
                        *(LOOP BACK CONNECTOR INSTALLED)
                        *
                        *TEST MODEM SIGNAL CCITT 107 AND CCITT 108 ON
                        *CHANEL B  WITH EXTERNAL LOOPBACK
                        *(LOOP BACK CONNECTOR INSTALLED)
                        *
                        ********************************************************



                        ********************** TEST 12 ********************
                        *
                        *TEST MODEM SIGNAL CCITT 105,106,109 ON CHANEL A
```

```
                                * WITH EXTERNAL LOOPBACK
                                *(LOOP BACK CONNECTOR INSTALLED)
                                *
                                *TEST MODEM SIGNAL CCITT 105,106,109 ON CHANEL B
                                * WITH EXTERNAL LOOP BACK
                                *(LOOP BACK CONNECTOR INSTALLED)
                                *
                                ************************************************
```

```
                                8.0 ERROR INFORMATION


                                8.1 ERROR REPORTING

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









                                9.0     HISTORY

                                   -    DESIGN STARTED ON JANUARY 82
                                   -    REV A0 ON DECEMBER 82          ;DIAGNOSTIC REVIEW
                                   -    REV B0 ON APRIL 83             ;WRONG CLOCK DIVIDER VALUE IN TWO TEST
                                a
```

```
                                .TITLE KMV11 B LINE CNT  DIAG.
002000                          .=2000



                                .MCALL  SVC
002000                          SVC                             ; INITIALIZE SUPERVISOR MACROS


002000                          BGNMOD  KMV11B

000000                  $LSTIN= 0
000000                  $LSTTAG= 0
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,BGNDU,BGNSETUP



002000                                  HEADER  VKMEB0,B,0,240.,0
```

```
1049
1050                                    ;++
1051                                    ; THIS TABLE IS USED BY THE RUNTIME SERVICES
1052                                    ; TO PROTECT THE LOAD MEDIA.
1053                                    ;--
1054
1055 002122                                    BGNPROT
1056
1057 002122  000000                            0                ;OFFSET INTO P-TABLE FOR CSR ADDRESS
1058 002124  177777                            -1               ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
1059 002126  177777                            -1               ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
1060
1061
1075
1076
1077 002130                                    ENDPROT
1078
```

```
                                .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 12
```

```
                                .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
                                        BGNHW   DFPTBL
002162
                                .WORD   177000          ;KMV11,CSRS ADDRESS
002164  177000                  .WORD   300             ;KMV11, VECTOR ADDRESS
002166  000300                  .WORD   4000            ;INTERRUPT PRIORITY LEVEL
002170  004000
```

```
   1127 002172  000001                              .WORD   1
   1128 002174                                     ENDHW
   1130
   1131
   1132
   1133
   1134
   1135
```

```
                                .SBTTL  GLOBAL EQUATES SECTION


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




002174                                  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
                                ;
                                ;
```

GLOBAL EQUATES SECTION

```
                                ; 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
1176
1177
1178    000340                  MAXPRI==340
1179    054000                  MAINT0==54000           ;MASTER CLEAR = 1,MODE = 1 ,MAINT 1 = 1 ,T11=HOLD
1180    044000                  MAINT1==44000           ;MASTER CLEAR = 1,MODE = 0 ,MAINT 1 = 0 ,T11=NOT HOLD
1181    040000                  MCLR==40000
1182    052525                  DATA1== 052525
1183    125252                  DATA2== 125252
1184    013224                  KB1.2== 5780.                   ;OCTAL VALUE OF 1,2 KBAUDS
1185    000154                  KB64==  108.                    ;  ''      ''    '' 64    ''
1186    000174                  KB56==  124.                    ;  ''      ''      56     ''
1187    000146                  KB68==  102.                    ;  ''      ''      68     ''
1188    000143                  KB70==  99.                     ;  ''      ''      70     ''
1189    000141                  KB72==  97.                     ;  ''      ''      72     ''
1190
1191                            ;                       DIVIDER CALCULATION= DECIMAL VALUE=6912 / X KBAUDS
1192
1193
1194
1195                            ;;*************************************************************************
1196                            ;* PROGRAM EVENT FLAG DEFINITIONS
1197                            ;;*************************************************************************
1198
1199
1200
```

```
                                .SBTTL  GLOBAL DATA SECTION

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



                                ;;**************************************************************************************
                                ;* STORAGE FOR DEVICE REGISTERS
                                ;;**************************************************************************************
002174                                  DESCRIPT        <KMV11B LINE CNT DIAG.>



                                        ERRTBL
002222  000000                  ERRTYP::        .WORD   0
002224  000000                  ERRNBR::        .WORD   0
002226  000000                  ERRMSG::        .WORD   0
002230  000000                  ERRBLK::        .WORD   0


                                ;**************************************************************************
                                ;* PROGRAM CONTROL PARAMETERS
                                ;**************************************************************************
002232  000000                  LOCK:   .WORD   0               ;ADDRESS FOR LOCK CURRENT DATA
002234  000005                  MAXERR: .WORD   5               ;MAX ERROR BEFORE DROPPING UNIT
002236  000000                  ERRCNT: .WORD   0               ;ERROR COUNT
002240  000000                  LOGDEV: .WORD   0               ;LOGICAL DEVICE NUMBER
002242  000000                  PSTACK: .WORD   0               ;BASE LEVEL PROGRAM STACK POINTER
002244  000000                  SAVSP:  .WORD   0               ;STACK POINTER STORAGE
002246  000000                  SAVPC:  .WORD   0               ;PROGRAM COUNTER STORAGE
002250  000000                  FLAG:   .WORD   0               ;SCRATCH STORAGE
002252  000000                  FTIME:  .WORD   0
002254  000000                  SAVE4:  .WORD   0
002256  000000                  SAVE6:  .WORD   0
002260  000000                  L$SW:   .WORD   0
002262  000015                  L$UIT:  .WORD   15
002264  000000                  UNIT:   .WORD   0
```

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

002266  000000          DELCT1: .WORD   0
002270  000000          DELCT2: .WORD   0
002272  000000          GOOD:   .WORD   0
002274  000000          GOOD0:  .WORD   0
002276  000000          GOOD1:  .WORD   0
002300  000000          GOOD2:  .WORD   0
002302  000000          GOOD4:  .WORD   0
002304  000000          GOOD6:  .WORD   0
002306  000000          GOOD10: .WORD   0
002310  000000          GOOD12: .WORD   0
002312  000000          GOOD14: .WORD   0
002314  000000          GOOD16: .WORD   0
002316  000000          SEL0:   .WORD   0
002320  000000          SEL1:   .WORD   0
002322  000000          SEL2:   .WORD   0
002324  000000          SEL4:   .WORD   0
002326  000000          SEL6:   .WORD   0
002330  000000          SEL10:  .WORD   0
002332  000000          SEL12:  .WORD   0
002334  000000          SEL14:  .WORD   0
002336  000000          SEL16:  .WORD   0
002340  000000          BSEL1:  .WORD   0
002342  000000          RANST:  .WORD   0
002344  000000          RANSEL: .WORD   0
002346  000000          RANMTA: .WORD   0
002350  000000          RANDN:  .WORD   0
002352  000000          SAVPC1: .WORD   0
002354  000000          SAVSTA: .WORD   0
002356  000000          COUNT:  .WORD   0
002360  000000          NUMBER: .WORD   0
002362  000000          ADDR:   .WORD   0
002364  000000          GDDAT:  .WORD   0
002366  000000          BDDAT:  .WORD   0

002370                  TTABLE: .BLKW   2000
006370                  RTABLE: .BLKW   2000

012370  000000          EXADDR: .WORD   0
012372  000000          INTFLG: .WORD   0
012374  000000          BAD:    .WORD   0
012376  000000          BSEL0:  .WORD   0
012400  000000          DATA:   .WORD   0
012402  000000          VECT:   .WORD   0


012404  000000          KIND:   .WORD   0               ;=0 IF KMV11A ,=1 IF KMV11B
012406  000000          CHANEL: .WORD   0

012410  000000          TXDATA: .WORD   0
012412  000000          RXDATA: .WORD   0
012414  000000          TSPEED: .WORD   0
012416  000000          LENGTH: .WORD   0
012420  000000          NUB:    .WORD   0
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 26-1
GLOBAL DATA SECTION

   1316 012422  000000                    RXCNT:  .WORD   0
   1317 012424  000000                    STAERR: .WORD   0
   1318 012426  000000                    WRDCNT: .WORD   0
```

```
1320
1321                            ;*****************************************************************
1322                            ;LOAD IN LOCATION "GDREV" THE PROM VERSION NUMBER THAT IS       *
1323                            ;COMPATIBLE WITH THIS DIAGNOSTIC                               *
1324                            ;                                                              *
1325                            ;EACH PROM CONTAIN A REV LEVEL AND A ECO LEVEL:                 *
1326                            ;THE REV LEVEL IS MODIFIED EACH TIME A MODIFICATION IS DONE     *
1327                            ;THE ECO LEVEL IS MODIFIED WHEN THE PROM MODIFICATION NEED      *
1328                            ;A DIAGNOSTIC MODIFICATION                                     *
1329                            ;*****************************************************************
1330
1331
1332
1333
1334
1335 012430  000001             GDREV:  .WORD   1
1336
1337
1338
1339
1340
```

```
1342                                ;*****************************************************************************
1343                                ;* PROGRAM CONTROL FLAGS
1344                                ;*****************************************************************************
1345 012432      000               INIFLG: .BYTE   0               ;PROGRAM INITIALIZING FLAG
1346                                       .EVEN
1347 012434      000               LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
1348 012435      000               QV.FLG: .BYTE   0               ;QUICK VERIFY FLAG
1349                                       .EVE
1350 012436  000000                UUT:    .WORD   0               ;CURRENT UNIT UNDER TEST
1351
1352
1353
1354
1355
1356                                ;*****************************************************************************
1357                                ;* POINTERS TO KMV11 VECTORS AND REGISTERS
1358                                ;*****************************************************************************
1359 012440  000000                KMVV00: 0       ;POINTER TO KMV11  INTRPT VECTOR 0
1360 012442  000000                KMVLVL: 0       ;POINTER TO KMV11  INTRPT SERVICE
1361 012444  000000                KMVV04: 0       ;POINTER TO KMV11  INTRPT VECTOR 04
1362 012446  000000                KMVV02: 0       ;    ''       ''       ''       ''     02
1363 012450  000000                KMVV06: 0       ;    ''       ''       ''       :     06
1364 012452  000000                KMTLVL: 0       ;POINTER TO KMV11 TX INTRPT SERVICE PS
1365 012454  000000                KMVCSR: 0       ;POINTER TO KMV11 CONTROL STATUS REGISTER
1366 012456  000000                KMVP02: 0       ;POINTER TO KMV11 PORT REGISTER - SEL2
1367 012460  000000                KMVP04: 0       ;POINTER TO KMV11 PORT REGISTER - SEL4
1368 012462  000000                KMVP06: 0       ;POINTER TO KMV11 PORT REGISTER - SEL6
1369
1370 012464  000000                KMVP10: 0       ;POINTER TO KMV11 PORT REG  -SEL10
1371 012466  000000                KMVP12: 0       ;POINTER TO PORT REG  -SEL 14
1372 012470  000000                KMVP14: 0       ;POINTER TO PORT REG  -SEL14
1373 012472  000000                KMVP16: 0       ;POINTER TO PORT REG 16
1374
1375 012474  000000                LOOP:   0       ;POINTER TO LOOP BACK CONNECTOR
1376
1377
1378                                ;;****  PRIMARY REG ADRS STORAGE FOR THIS UNIT  ******
1379                                ;THESE LOCATIONS WILL BE LOADED FOR THE CURRENT UNIT, IN INIT CODE
1380 012476                        REGADR:
1381
1382                                ;;****  STACK USED FOR SUBROUTINE LINKAGE  *****
1383 012476                                .BLKW   100
1384 012676                        SSTACK:
1385
1386
1387
1388
1389
1390
1391
```

```
                .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
                ;;*****************************************************************************
012676                  DEVTYP  <KMV11B>


                ;
                ; FORMAT STATEMENTS USED IN PRINT CALLS
                ;
```

```
        .SBTTL  GLOBAL SUBROUTINES




;-------------------------------------------------------------------
; MACRO'S NEEDED TO CALL SUBROUTINES
;-------------------------------------------------------------------

.MACRO  CLRMAR
        ROMCLK
        004000
.ENDM   CLRMAR
```

GLOBAL SUBROUTINES

```
                                        ;ROUTINE TO WAIT FOR EVENT OR TIMEOUT


                                        ;CALLING SEQUENCE:      JSR   PC,WAIT1
                                        ;                       JSR   PC,WAIT2


                                        ;INPUTS PARAMETERS:     DELCT1,DELCT2

                                        ;                       INC DELCT1 UNTIL 0
                                        ;                       DEC DELCT2 UNTIL 0      DELCT2= NUMB OF WAIT1 PASSES




 012706  005237  002266        WAIT2:  INC     DELCT1
 012712  001375                        BNE     WAIT2

 012714                                 BREAK

 012716  005337  002270                DEC     DELCT2
 012722  001371                        BNE     WAIT2

 012724  000207                        RTS     PC




 012726  005237  002266        WAIT1:  INC     DELCT1
 012732  001375                        BNE     WAIT1
 012734                                 BREAK

 012736  000207                        RTS     PC
```

```
1479                                    ;MACRO TO WAIT A FEW MS
1480
1481
1482                                    ;CALLING SEQUENCE:      WAITA   X               0<X<177777
1483                                    ;                       WAITB   X,Y             0<X OR Y<177777
1484
1485
1486
1487                                    .MACRO  WAITA   X
1488                                            MOV     #X,DELCT1               ;LOAD COUNT
1489                                            JSR     PC,WAIT1                ;WAIT
1490                                    .ENDM
1491
1492
1493
1494
1495
1496
1497
1498                                    .MACRO  WAITB   X,Y
1499                                            MOV     #X,DELCT1
1500                                            MOV     #Y,DELCT2
1501                                            JSR     PC,WAIT2
1502                                    .ENDM
1503
1504
```

```
1506
1507                                    ;ROUTINE TO DROP UNIT AFTER 5 ERROR
1508
1509
1510                                    ;JSR   PC,CHKMAX
1511
1512
1513
1514
1515
1516
1517
1518
1519 012740                            CHKMAX: INLOOP                          ;LOOPING ON ERROR?
1520 012742                                    BCOMPLETE       1$              ;IF YES, EXIT
1521
1522
1523 012744                                    RFLAGS  R0                      ;GET OPERATOR FLAG
1524 012746  032700  000040                    BIT     #IDU,R0                 ;IS DROPPING INHIBITTED?
1525 012752  001026                            BNE     1$                      ;IF YES EXIT
1526
1527
1528 012754  005237  002236                    INC     ERRCNT                  ;UPDATE ERROR COUNT
1529 012760  023737  002236  002234            CMP     ERRCNT,MAXERR           ;TOO MANY ERROR?
1530 012766  003420                            BLE     1$                      ;IF NOT JUMP
1531
1532
1533 012770                                    PRINTF  #NERRS,MAXERR,UUT       ;TOO MANY ERROR!
1534 013020                                    DODU    UUT                     ;DROP UNIT
1535
1536 013026                                    DOCLN                           ;END THE SUBPASS
1537
1538 013030  000207                    1$:     RTS     PC
1539
1540
1541
1542
1543                                           .NLIST  BEX
1544 013032      045      116      045  NERRS: .ASCIZ  /%N?AMORE THAN %D3%A  ERRORS ON UNIT %D2/
1545                                           .LIST   BEX
1546                                           .EVEN
1547
1548
1549
1550
1551
```

```
                                ;ROUTINE TO CHECK REGISTER BSEL0 AND TO REPORT ERROR
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;CALLING SEQUENCE:      JSR     PC,TSTERR
                                ;
                                ;
                                ;OUTPUT PARAMETERS:     RETURN TO       PC      IF TEST IS OK
                                ;                                       PC+2    IF TIMEOUT DURING TEST
                                ;                                       PC+4    IF NO KMV11 ANSWER
                                ;                                       PC+6    IF DATA CMP ERROR
                                ;
                                ;
                                ;
                                ;
013102  004537  013652  TSTERR: JSR     R5,CBSEL0               ;LOOK IF BSEL0=0
013106  000000                 .WORD   0
013110  000411                 BR      1$                      ;TEST IS OK ,RTS PC
                                ;
                                ;
013112  004537  013652          JSR     R5,CBSEL0               ;LOOK IF BSEL0=200
013116  000200                 .WORD   200
013120  000406                 BR      2$                      ;TIMEOUT DURING TEST,RTS PC+2
                                ;
                                ;
013122  004537  013652          JSR     R5,CBSEL0               ;LOOK IF BSEL0=100
013126  000100                 .WORD   100
013130  000405                 BR      3$                      ;DATA CMP ERROR,RTS PC+6
                                ;
                                ;
                                ;
013132  000407                 BR      4$                      ;NO KMV11 ANSWER ,RTS PC+4
                                ;
                                ;
                                ;
013134  000207          1$:     RTS     PC                      ;TEST OK
                                ;
                                ;
013136  062716  000002  2$:     ADD     #2,(SP)
013142  000207                 RTS     PC                      ;TIMEOUT ERROR
                                ;
                                ;
013144  062716  000006  3$:     ADD     #6,(SP)
013150  000207                 RTS     PC                      ;DATA CMP ERROR
                                ;
                                ;
013152  062716  000004  4$:     ADD     #4,(SP)
013156  000207                 RTS     PC                      ;NO KMV11 ANSWER
```

```
1608                                    .SBTTL  NUMBER GENERATOR
1609                                ;
1610                                ;
1611                                ;
1612                                ;       DESCRIPTION:
1613                                ;
1614                                ;               ROUTINE TO GENERATE DATA PATTERNS,
1615                                ;       THE TYPE OF PATTERN IS SELECTED BY R3, AND THE
1616                                ;       PATTERN GENERATED IS RETURNED IN LOCATION 'DATA"
1617                                ;       AND LOCATION "GOOD"
1618                                ;
1619                                ;       CALLING SEQUENCE:
1620                                ;
1621                                ;               JSR     PC,GENER
1622                                ;
1623                                ;       INPUT PARAMETERS:
1624                                ;
1625                                ;       R3 CONTAINS THE PATTERN NUMBER
1626                                ;
1627                                ;       R3=0            ALL ZEROES
1628                                ;       1               ALL ONES
1629                                ;       2               010101 ETC BIT PATTERN
1630                                ;       3               101010 ETC BIT PATTERN
1631                                ;       4               ROTATING 1 IN A ZERO WORD
1632                                ;       5               ROTATING 0 IN AN ALL ONE WORD
1633                                ;       6               PSEUDO RANDOM NUMBER
1634                                ;       7               INCREMENTING DATA PATTERN, GOOD
1635                                ;                       CONTAINS THE VALUE TO BE UPDATED
1636                                ;
1637                                ;
1638                                ;
1639                                ;
1640                                ;       IMPLICIT INPUT PARAMETERS:
1641                                ;
1642                                ;               NONE
1643                                ;
1644                                ;
1645                                ;       OUTPUT PARAMETERS:
1646                                ;
1647                                ;               THE NUMBER GENERATED IS HELD IN
1648                                ;       DATA AND GOOD.
1649                                ;
1650                                ;
1651                                ;       IMPLICIT OUTPUT PARAMETERS:
1652                                ;
1653                                ;               NONE
1654                                ;
1655                                ;
1656                                ;       COMPLETION CODES:
1657                                ;
1658                                ;               NONE
1659                                ;
1660                                ;
1661                                ;       POSSIBLE ERROR CODES:
1662                                ;
1663                                ;               NONE
1664                                ;
```

```
1665                                    ;
1666                                    ;
1667 013160  042703  177770            GENER:  BIC     #177770,R3
1668 013164  004737  013460                    JSR     PC,SAVREG
1669 013170  006303                            ASL     R3
1670 013172  000173  013176                    JMP     @GENSEL(R3)
1671 013176  013216                    GENSEL: GEN0                    ;ALL ZERO WORD
1672 013200  013222                            GEN1                    ;ALL ONE WORD
1673 013202  013230                            GEN52                   ;52 PATTERN
1674 013204  013236                            GEN25                   ;25 PATTERN
1675 013206  013244                            GENR1                   ;ROTATE '1' EACH CALL
1676 013210  013254                            GENR0                   ;ROTATE '0' EACH CALL
1677 013212  013312                            GENRAN                  ;RANDOM NUMBER
1678 013214  013432                            GENINC                  ;INCREMENTING COUNT
1679 013216  005000                    GEN0:   CLR     R0              ;0>R0
1680 013220  000507                            BR      GENEX
1681 013222  005000                    GEN1:   CLR     R0              ;NOT0>R0
1682 013224  005100                            COM     R0
1683 013226  000504                            BR      GENEX
1684 013230  012700  052525            GEN52:  MOV     #52525,R0       ;5252>R0
1685 013234  000501                            BR      GENEX
1686 013236  012700  125252            GEN25:  MOV     #125252,R0      ;125252>R0
1687 013242  000476                            BR      GENEX
1688 013244  000241                    GENR1:  CLC
1689 013246  004737  013266                    JSR     PC,GENROT       ;SHIFT 1 > R0
1690 013252  000472                            BR      GENEX
1691 013254  000241                    GENR0:  CLC
1692 013256  004737  013266                    JSR     PC,GENROT       ;
1693 013262  005100                            COM     R0              ;SHIFT 0 > R0
1694 013264  000465                            BR      GENEX
1695 013266  006037  013310            GENROT: ROR     GENISH          ;ROTATE  1 PATTERN
1696 013272  001003                            BNE     GENER1          ;= 0?
1697 013274  012737  100000  013310            MOV     #100000,GENISH  ;YES, SET MSB
1698 013302  013700  013310            GENER1: MOV     GENISH,R0       ;PUT 1 IN R0
1699 013306  000207                            RTS     PC              ;AND EXIT
1700 013310  000001                    GENISH: 1
1701 013312  012737  000005  002344    GENRAN: MOV     #5,RANSEL       ;SET SELECT VALUE TO 5
1702 013320  004737  013332                    JSR     PC,RANGEN       ;GENERATE RANDOM NUMBER IN R0
1703 013324  013700  002350                    MOV     RANDN,R0        ;
1704 013330  000443                            BR      GENEX           ;
1705 013332  013702  002350            RANGEN: MOV     RANDN,R2        ;
1706 013336  001002                            BNE     RAN1            ;IS RANDOM = 0
1707 013340  013702  002342                    MOV     RANST,R2        ;YES, PUT RANDOM START VALUE IN
1708 013344  032737  000777  002344    RAN1:   BIT     #777,RANSEL     ;NO;IS RANSEL SELECT VALUE = 0
1709 013352  001003                            BNE     RAN2            ;NO
1710 013354  012737  000001  002344            MOV     #1,RANSEL       ;YES: SET RANSEL = 1
1711 013362  013703  002344    RAN2:   MOV     RANSEL,R3       ;
1712 013366  013702  002350            MOV     RANDN,R2
1713 013372  033702  002346            BIT     RANMTA,R2       ;GET R2 <0 AND 1>
1714 013376  001405                    BEQ     RANCLC          ;
1715 013400  005102                    COM     R2
1716 013402  033702  002346            BIT     RANMTA,R2
1717 013406  001401                    BEQ     RANCLC
1718 013410  000402                    BR      RANSEC
1719 013412  000241            RANCLC: CLC
1720 013414  000401                    BR      RAN4
1721 013416  000261            RANSEC: SEC
```

```
1722 013420  006037  002350          RAN4:   ROR     RANDN           ;ROTATE C TO B15
1723 013424  005303                          DEC     R3              ;IS THIS NUMBER REQUIRED?
1724 013426  001357                          BNE     RAN2+4          ;NO, GET ANOTHER
1725 013430  000207                  RANEX:  RTS     PC              ;YES, EXIT
1726 013432  013700  002272          GENINC: MOV     GOOD,R0         ;INCREMENTS LOC. 'GOOD'
1727 013436  005200                          INC     R0
1728 013440  010037  002272          GENEX:  MOV     R0,GOOD
1729 013444  004737  013540                  JSR     PC,RSTREG
1730 013450  013737  002272  012400          MOV     GOOD,DATA
1731 013456  000207                          RTS     PC
1732
```

```
                        .SBTTL  SAVE REGISTERS
                ;
                ;
                ;       DESCRIPTION:
                ;
                ;               ROUTINE TO SAVE ALL THE GENERAL PURPOSE
                ;       REGISTERS ON THE STACK, AND LEAVE THE ADDRESS OF THE
                ;       CALLING ROUTINE ON THE STACK. THE ROUTINE WILL RUN AT
                ;       PRIORITY 7 TO AVOID ANY INTERRUPTS
                ;
                ;       CAUTION:REGISTER R0 IS NOT SAVED
                ;
                ;
                ;       CALLING SEQUENCE:
                ;
                ;               JSR     PC,SAVREG
                ;
                ;       INPUT PARAMETERS:
                ;
                ;               NONE
                ;
                ;
                ;       IMPLICIT INPUT PARAMETERS:
                ;
                ;               NONE
                ;
                ;       OUTPUT PARAMETERS:
                ;
                ;               REGISTERS 0 THRU 5 ARE SAVED ON THE STACK
                ;       AND THE RETURN ADDRESS OF THE CALLING ROUTINE IS
                ;       SET AS THE LAST ENTRY ON THE STACK
                ;
                ;
                ;       IMPLICIT OUTPUT PARAMETERS:
                ;
                ;               NONE
                ;
                ;       COMPLETION CODES:
                ;
                ;               NONE
                ;
                ;       POSSIBLE ERROR CODES:
                ;
                ;               NONE
                ;
013460          SAVREG: GETPRI  SAVSTA
013466                  SETPRI  MAXPRI
013474  012637  002246          MOV     (SP)+,SAVPC     ;SAVE PC FOR RETURN FROM THIS ROUTINE
```

SAVE REGISTERS

```
1791 013500  012637  002352                MOV     (SP)+,SAVPC1
1792 013504  010546                        MOV     R5,-(SP)
1793 013506  010446                        MOV     R4,-(SP)
1794 013510  010346                        MOV     R3,-(SP)
1795 013512  010246                        MOV     R2,-(SP)
1796 013514  010146                        MOV     R1,-(SP)
1797 013516  010046                        MOV     R0,-(SP)
1798 013520  013746  002352                MOV     SAVPC1,-(SP)
1799 013524  013746  002246                MOV     SAVPC,-(SP)     ;PUT PC READY FOR
1800 013530                                SETPRI  SAVSTA
1801 013536  000207                        RTS     PC              ;RETURN
1802
1803
1804
```

```
                                        .SBTTL  RESTORE REGISTERS
                                ;
                                ;
                                ;       DESCRIPTION:
                                ;
                                ;               RESTORE TO RESTORE THE GENERAL PURPOSE
                                ;       REGISTERS. THE STACK IS LEFT IN THE SAME STATE AS IT
                                ;       WAS WHEN SAVREG WAS CALLED.
                                ;
                                ;       CAUTION:  REGISTER R0 IS NOT SAVED
                                ;
                                ;
                                ;
                                ;       CALLING SEQUENCE:
                                ;
                                ;               JSR     PC,RSTREG
                                ;
                                ;       INPUT PARAMETERS:
                                ;
                                ;               NONE
                                ;
                                ;       IMPLICIT INPUT PARAMETERS:
                                ;
                                ;               NONE
                                ;
                                ;       OUTPUT PARAMETERS:
                                ;
                                ;               R1 THRU R5 RESTORED
                                ;
                                ;       IMPLICIT OUTPUT PARAMETERS:
                                ;
                                ;               NONE
                                ;
                                ;       COMPLETION CODES:
                                ;
                                ;               NONE
                                ;
                                ;       POSSIBLE ERROR CODES:
                                ;
                                ;               NONE
                                ;
                                ;
013540                          RSTREG: GETPRI  SAVSTA
013546                                  SETPRI  MAXPRI
013554  012637  002246                  MOV     (SP)+,SAVPC
013560  012637  002352                  MOV     (SP)+,SAVPC1
013564  012600                          MOV     (SP)+,R0
013566  012601                          MOV     (SP)+,R1
```

RESTORE REGISTERS

```
   1863 013570  012602                          MOV     (SP)+,R2
   1864 013572  012603                          MOV     (SP)+,R3
   1865 013574  012604                          MOV     (SP)+,R4
   1866 013576  012605                          MOV     (SP)+,R5
   1867 013600  013746  002352                  MOV     SAVPC1,-(SP)
   1868 013604  013746  002246                  MOV     SAVPC,-(SP)     ;PUT PC READY FOR
   1869 013610                                  SETPRI  SAVSTA
   1870 013616  000207                          RTS     PC
```

```
                                        ;CHECK CONTENT OF ONE OF THE 8 REGISTERS
                                        ;
                                        ; CALLING SEQUENCE
                                        ;       JSR     R5,CKSELN               ; N = REGISTER NUMBER
                                        ;       .WORD A                 A=EXPECTED CONTENT OF REGISTER N
                                        ;
                                        ;OUTPUT PARAMETER:
                                        ;       BRANCH IN PC+2 IF ERROR DETECTED
                                        ;       BRANCH IN PC IF NO ERROR DETECTED
                                        ;
                                        ;
                                        ;
                                        ;
013620  012537  002272          CKSEL0: MOV     (R5)+,GOOD      ;WRITE GOOD
013624  017737  176624  002316          MOV     @KMVCSR,SEL0    ;READ SEL 0
013632  023737  002316  002272          CMP     SEL0,GOOD       ;CMP ?
013640  001001                          BNE     1$
013642  000402                          BR      2$
013644  062705  000002          1$:     ADD     #2,R5
013650  000205                  2$:     RTS     R5
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
013652  005037  002272          CBSEL0: CLR     GOOD
013656  012537  002272                  MOV     (R5)+,GOOD
013662  117737  176566  012376          MOVB    @KMVCSR,BSEL0
013670  123737  012376  002272          CMPB    BSEL0,GOOD
013676  001001                          BNE     1$
013700  000402                          BR      2$
013702  062705  000002          1$:     ADD     #2,R5
013706  000205                  2$:     RTS     R5
```

RESTORE REGISTERS

```
                                        ;ROUTINE TO CHECK ALL REGISTER FROM SEL0 TO SEL16
                                        ;
                                        ;CALLING SEQUENCE:
                                        ;       JSR R5,CKALL
                                        ;       .WORD A                 A = EXPECTED VALUE FOR SEL0
                                        ;       .WORD B                 B       "       "      SEL2
                                        ;       .WORD C                 C       "       "      SEL4
                                        ;       .WORD D                 D       "       "      SEL6
                                        ;       .WORD E                 E       "       "      SEL10
                                        ;       .WORD F                 F       "       "      SEL12
                                        ;       .WORD G                 G       "       "      SEL14
                                        ;       .WORD H                 H       "       "      SEL16
                                        ;
                                        ;OUTPUT PARAMETER:
                                        ;       BRANCH IN PC+2 IF ERROR
                                        ;       BRANCH IN PC IF NO ERROR
                                        ;
                                        ;
013710  012537  002274          CKALL:  MOV     (R5)+,GOOD0
013714  012537  002300                  MOV     (R5)+,GOOD2
013720  012537  002302                  MOV     (R5)+,GOOD4
013724  012537  002304                  MOV     (R5)+,GOOD6
013730  012537  002306                  MOV     (R5)+,GOOD10
013734  012537  002310                  MOV     (R5)+,GOOD12
013740  012537  002312                  MOV     (R5)+,GOOD14
013744  012537  002314                  MOV     (R5)+,GOOD16
                                        ;
013750  017737  176500  002316          MOV     @KMVCSR,SEL0            ;READ SEL0
013756  000240                          NOP
013760  017737  176472  002322          MOV     @KMVP02,SEL2            ;READ SEL2
013766  000240                          NOP
013770  017737  176464  002324          MOV     @KMVP04,SEL4            ;READ SEL4
013776  000240                          NOP
014000  017737  176456  002326          MOV     @KMVP06,SEL6            ;READ SEL6
014006  000240                          NOP
014010  017737  176450  002330          MOV     @KMVP10,SEL10           ;READ SEL10
014016  000240                          NOP
014020  017737  176442  002332          MOV     @KMVP12,SEL12           ;READ SEL12
014026  000240                          NOP
014030  017737  176434  002334          MOV     @KMVP14,SEL14           ;READ SEL14
014036  000240                          NOP
014040  017737  176426  002336          MOV     @KMVP16,SEL16           ;READ SEL16
                                        ;
014046  023737  002316  002274          CMP     SEL0,GOOD0
014054  001035                          BNE     1$
014056  023737  002322  002300          CMP     SEL2,GOOD2
014064  001031                          BNE     1$
014066  023737  002324  002302          CMP     SEL4,GOOD4
014074  001025                          BNE     1$
014076  023737  002326  002304          CMP     SEL6,GOOD6
014104  001021                          BNE     1$
014106  023737  002330  002306          CMP     SEL10,GOOD10
014114  001015                          BNE     1$
014116  023737  002332  002310          CMP     SEL12,GOOD12
```

RESTORE REGISTERS

```
   1966 014124  001011                          BNE     1$
   1967 014126  023737  002334  002312          CMP     SEL14,GOOD14
   1968 014134  001005                          BNE     1$
   1969 014136  023737  002336  002314          CMP     SEL16,GOOD16
   1970 014144  001001                          BNE     1$
   1971
   1972 014146  000402                          BR      2$
   1973 014150  062705  000002          1$:     ADD     #2,R5
   1974 014154  000205                  2$:     RTS     R5
```

RESTORE REGISTERS

```
                                        ;ROUTINE TO CHECK SEL2 TO SEL16


014156  012537  002300          CKREG:  MOV     (R5)+,GOOD2
014162  012537  002302                  MOV     (R5)+,GOOD4
014166  012537  002304                  MOV     (R5)+,GOOD6
014172  012537  002306                  MOV     (R5)+,GOOD10
014176  012537  002310                  MOV     (R5)+,GOOD12
014202  012537  002312                  MOV     (R5)+,GOOD14
014206  012537  002314                  MOV     (R5)+,GOOD16

014212  017737  176240  002322          MOV     @KMVP02,SEL2
014220  000240                          NOP
014222  017737  176232  002324          MOV     @KMVP04,SEL4
014230  000240                          NOP
014232  017737  176224  002326          MOV     @KMVP06,SEL6
014240  000240                          NOP
014242  017737  176216  002330          MOV     @KMVP10,SEL10
014250  000240                          NOP
014252  017737  176210  002332          MOV     @KMVP12,SEL12
014260  000240                          NOP
014262  017737  176202  002334          MOV     @KMVP14,SEL14
014270  000240                          NOP
014272  017737  176174  002336          MOV     @KMVP16,SEL16


014300  023737  002322  002300          CMP     SEL2,GOOD2
014306  001031                          BNE     1$
014310  023737  002324  002302          CMP     SEL4,GOOD4
014316  001025                          BNE     1$
014320  023737  002326  002304          CMP     SEL6,GOOD6
014326  001021                          BNE     1$
014330  023737  002330  002306          CMP     SEL10,GOOD10
014336  001015                          BNE     1$
014340  023737  002332  002310          CMP     SEL12,GOOD12
014346  001011                          BNE     1$
014350  023737  002334  002312          CMP     SEL14,GOOD14
014356  001005                          BNE     1$
014360  023737  002336  002314          CMP     SEL16,GOOD16
014366  001001                          BNE     1$
014370  000402                          BR      2$

014372  062705  000002          1$:     ADD     #2,R5
014376  000205                  2$:     RTS     R5
```

RESTORE REGISTERS

```
                                        ;ROUTINE TO CLEAR KMV11 MODULE
                                        ;
                                        ;CALLING SEQUENCE:
                                        ;       JSR PC,CLRKMV
                                        ;
                                        ;ROUTINE DESCRIPTION: CLEAR ALL CSR'S REGISTERS AND CHECK IF = 0
                                        ;
                                        ;
                                        CLRKMV:
014400  005077  176050                          CLR     @KMVCSR                 ;CLEAR REGISTETRS
014404  012777  054000  176042                  MOV     #MAINTO,@KMVCSR         ;SET MAINTENANCE MODE
014412                                          WAITA   0
                                        ;
                                        ;
014424  012702  000010                          MOV     #10,R2
014430  013701  012454                          MOV     KMVCSR,R1               ;LOAD ADDRESS
014434  005021                          1$:     CLR     (R1)+                   ;CLEAR
014436  005302                                  DEC     R2                      ;ALL DONE
014440  001375                                  BNE     1$                      ;NO
014442  004537  013710                          JSR     R5,CKALL                ;CHECK ALL REG = 0
014446  000000                                  .WORD   0
014450  000000                                  .WORD   0
014452  000000                                  .WORD   0
014454  000000                                  .WORD   0
014456  000000                                  .WORD   0
014460  000000                                  .WORD   0
014462  000000                                  .WORD   0
014464  000000                                  .WORD   0
014466  000404                                  BR      2$                      ;OK BRANCH AT END
014470                                          ERRHRD 1,EM0002,PRALL           ;CSR'S REGISTERS CAN'T BE CLEARED
014500  000207                          2$:     RTS     PC
```

RESTORE REGISTERS

```
                                        ;ROUTINE TO SET MAINT MODE 1 AND CHECK DCT11 CLEAR SEL0 AFTER HAVING DECODED
                                        ;
                                        ;
                                        ;CALLING SEQUENCE:
                                        ;       JSR PC,MAINM1
                                        ;
                                        ;
                                        ;GIVE AN ERROR IF MASTER CLEAR IS NOT CLEAR BY DCT11
                                        ;
                                        ;MAINT1= MASTER CLEAR=1 + MAINT 1 =0 + MODE = 1 : T11=HOLD
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
014502  012777  044000  175744  MAINM1: MOV     #MAINT1,@KMVCSR         ;LOAD ADDRESS
014510  012737  000000  002266          MOV     #0,DELCT1
014516  012737  000001  002270          MOV     #1,DELCT2
014524  004737  012706                  JSR     PC,WAIT2
014530  004537  013620                  JSR     R5,CKSEL0               ;CHECK SEL0=0 BUT MODE BIT =1
014534  004000                          .WORD   4000
014536  000404                          BR      1$                      ;OK BRANCH
014540                                  ERRHRD  2,EM0001,PRSEL0
014550  000207                  1$:     RTS     PC
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
```

RESTORE REGISTERS

```
2097                                    ;ROUTINE TO SET TEST NUMBER ON BSEL0
2098
2099
2100
2101
2102                                    ;CALLING SEQUENCE:
2103                                    ;       JSR R5,TSTNUB
2104                                    ;       .WORD   A                       A=TEST MICRO PROGRAM NUMBER
2105
2106
2107
2108
2109
2110
2111 014552  012537  012420            TSTNUB: MOV     (R5)+,NUB
2112 014556  053777  012420  175670            BIS     NUB,@KMVCSR             ;LOAD TEST NUMBER
2113 014564  012737  000000  002266            MOV     #0000,DELCT1
2114 014572  004737  012726                    JSR     PC,WAIT1                ;WAIT
2115 014576  000205                            RTS     R5
```

```
                                                        ;ROUTINE TO CHECK IF KMV11A OR B AND IF LOOP BACK CONNECTOR
                                                        ;ARE PLUGGED OR NOT



                                                        ;CALLING SEQUENCE:

                                                        ;JSR    PC,CKKMV


                                                        ;OUTPUTS:       IF LOOP BACK    LOOP=1
                                                        ;               IF NO LOOP BACK         LOOP=0
                                                        ;
                                                        ;               IF      KMV11 A         KIND=0
                                                        ;               IF      KMV11 B         KIND=1



                                                        ;MICRO DIAG NB 44 DESCRIPTION:
                                                        ;DCT11 LOOK IF KMV11A OR B BY READING BIT 1 OF8255 CHIP PORT C
                                                        ;IF THIS BIT =0 IT IS KMV11 B MODULE AND DCT11 SET 1 IN BSEL4
                                                        ;     "   "   1  "    "   A   "        "   "  0   "
                                                        ;
                                                        ;DCT11 LOOK IF LOOP BACK OR NOT BY READING BIT 0 OF 8255 CHIP
                                                        ;IF THIS BIT =0 LOOP BACK CONNECTOR ARE PLUGGED SO DCT11 SET 1
                                                        ;IN BSEL2 IN OTHER CASE BSEL2=0
                                                        ;
                                                        ;NOTE:PORT C ADDRESS=13006
                                                        ;


014600  004737  014400           CKKMV:  JSR     PC,CLRKMV
014604  004737  014502                   JSR     PC,MAINM1       ;SET MAINT MODE
014610  004537  014552                   JSR     R5,TSTNUB       ;CHECK WHICH KMV11 AND IF LOOP BACK
014614  000044                           .WORD   44

014616  004737  013102                   JSR     PC,TSTERR       ;LOOK IF TEST CORRECTLY DONE
014622  000412                           BR      1$
014624  000402                           BR      2$
014626  000401                           BR      2$
014630  000400                           BR      2$



014632                           2$:     ERRHRD  3,EM0004        ;NO KMV11 ANSWER
014642  004737  012740                   JSR     PC,CHKMAX       ;LOOK IF MAX ERROR
014646  000207                           RTS     PC

014650  017737  175602  012474   1$:     MOV     @KMVP02,LOOP    ;WRITE LOOP BIT
014656  042737  177776  012474           BIC     #177776,LOOP
014664  017737  175570  012404           MOV     @KMVP04,KIND    ;WRITE KIND
014672  042737  177776  012404           BIC     #177776,KIND
```

RESTORE REGISTERS

```
   2174 014700  000207                          RTS     PC
```

```
2176
2177                                    ;ROUTIN TO WRITE OR READ ONE OF THE KMV11 REGISTERS
2178
2179
2180
2181                                    ;CALLING SEQUENCE:
2182                                    ;JSR     R5,WRITE
2183                                    ;.WORD   A                                  A=ADDRESS TO WRITE
2184                                    ;.WORD   B                                  B=DATA TO WRITE
2185                                    ;
2186                                    ;
2187                                    ;
2188                                    ;JSR     R5,READ
2189                                    ;.WORD   A                                  A=ADDRESS TO READ
2190                                    ;
2191                                    ;
2192                                    ;
2193                                    ;MICRO DIAG NB 47 DESCRIPTION:
2194                                    ;WRITE: PUT ADDRESS TO WRITE IN SEL2
2195                                    ;       PUT DATA TO WRITE IN SEL4
2196                                    ;       SET BIT 0 OF SEL6(WRITE BIT)
2197                                    ;       SET TEST NB 44
2198                                    ;       KMV11 CLEAR BSEL0 WHEN DONE
2199                                    ;
2200                                    ;
2201                                    ;READ:  PUT ADDRESS TO READ IN SEL2
2202                                    ;       CLEAR BIT 0 IN SEL6
2203                                    ;       SET TEST 47
2204                                    ;       KMV11 READ ADDRESS IN SEL2 AND CLEAR BSEL0 WHEN DONE
2205
2206
2207
2208
2209 014702  012577  175550          WRITE:  MOV     (R5)+,@KMVP02               ;WRITE ADDRESS
2210 014706  012577  175546                  MOV     (R5)+,@KMVP04               ;   "    DATA
2211 014712  012777  000001  175542          MOV     #1,@KMVP06                  ;BIT WRITE
2212
2213 014720  004537  014552                  JSR     R5,TSTNUB                   ;SEND TEST NB 47
2214 014724  000047                          .WORD   47
2215
2216 014726  000205                          RTS     R5                          ;RETURN
2217
2218
2219
2220
2221
2222
2223 014730  012577  175522          READ:   MOV     (R5)+,@KMVP02               ;SET ADDRESS TO READ
2224 014734  005077  175520                  CLR     @KMVP04
2225 014740  005077  175516                  CLR     @KMVP06
2226
2227 014744  004537  014552                  JSR     R5,TSTNUB                   ;SEND TEST NB 44
2228 014750  000047                          .WORD   47
2229
2230
2231 014752  004737  013102                  JSR     PC,TSTERR                   ;CHECK BSEL 0
2232 014756  000412                          BR      1$                          ;OK
```

RESTORE REGISTERS

```
 2233 014760  000402                          BR      2$
 2234 014762  000401                          BR      2$
 2235 014764  000400                          BR      2$
 2236
 2237 014766                          2$:      ERRHRD  4,EM0004              ;NO KMV ANSWER
 2238 014776  004737  012740                  JSR     PC,CHKMAX               ;MAX ERROR?
 2239 015002  000205                          RTS     R5
 2240
 2241 015004  017737  175450  012374  1$:     MOV     @KMVP04,BAD             ;READ DATA IN BAD
 2242 015012  000205                          RTS     R5
 2243
 2244
 2245
```

RESTORE REGISTERS

```
2247
2248
2249                            .MACRO  ROMCLK
2250                            .LIST
2251                                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
2252                            .NLIST
2253                            .ENDM
2254
2255                            .MACRO  ED$CALL XY
2256                                    .LIST
2257                            ;********************** TEST'XY' **************
2258                                    .NLIST
2259                                    .ENDM
2260
2261
2262
2263                            .MACRO  BADHEAD
2264                                    .RADIX  10
2265                                    ED$CALL \T$TESTNUM+1
2266                                    .RADIX  8
2267                                    .ENDM
2268
2269
2270
2271
```

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION

                                ;////////////////////////////////////////////////////////////////
                                ;/      THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
                                ;/      THAT ARE USED IN MORE THAN ONE TEST.
                                ;////////////////////////////////////////////////////////////////

                                .NLIST BEX

015014      040     102     125 TIM:    .ASCIZ  / BUS TIMEOUT /

015032      045     116     045 TFM36:  .ASCIZ  /%N%AREGISTER ADDRESS ERROR,ADDRESS = %06%A,UNIT = %02/

015120      115     101     123 EM0001: .ASCIZ  /MASTER CLEAR FAILS TO RESET: DCT11 CAN'T CLEAR MASTER CLEAR /

015215      040     113     115 EM0002: .ASCIZ  / KMV11 REGISTERS CANNOT BE CLEARED /

015261      040     104     101 EM0003: .ASCIZ  / DATA COMPARE ERROR ON KMV11 REGISTER (SEL2 TO SEL16)/

015347      040     116     117 EM0004: .ASCIZ  / NO ANSWER FROM KMV11 /

015376      040     124     111 EM0006: .ASCIZ  / TIMEOUT DURING KMV11 MICRO TEST /

015440      111     116     124 EM0007: .ASCIZ  /INTERRUPT OCCURED ON KMV11,WHEN ADDRESSING  CSR REGISTER/

015531      116     117     040 EM0010: .ASCIZ  /NO KMV11 INTERRUPT WHEN CSR'S ARE ACCESSED/

015604      113     115     126 EM0011: .ASCIZ  /KMV11 REAL LINE TIME CLOCK FAILED TO INTERRUPT /

015664      103     110     101 EM0012: .ASCIZ  /CHANEL A GENERATOR COUNT CANNOT BE READ OR WRITTEN CORRECTLY /

015762      103     110     101 EM0013: .ASCIZ  /CHANEL A  GENERATOR OUTPUT IS NOT CORRECT/

016034      103     110     101 EM0112: .ASCIZ  /CHANEL B GENERATOR COUNT CANNOT BE READ OR WRITTEN CORRECTLY /

016132      103     110     101 EM0113: .ASCIZ  /CHANEL B  GENERATOR OUTPUT IS NOT CORRECT/

016204      125     116     101 EM0033: .ASCIZ  /UNABLE TO CHANGE BAUD RATE GENERATOR COUNTER /

016262      116     117     040 EM0014: .ASCIZ  /NO CHANGE ON BAUD RATE GENERATOR OUTPUT /

016333      124     122     101 EM0015: .ASCIZ  /TRANSMITTION ERROR IN INTERNAL LOOP ON CH A WITHOUT INTERRUPTS/

016432      105     122     122 EM0016: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN INTERNAL LOOPBACK ON CH A /

016527      124     122     101 EM0115: .ASCIZ  /TRANSMITTION ERROR IN INTERNAL LOOP ON CH B WITHOUT INTERRUPTS/

016626      105     122     122 EM0116: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN INTERNAL LOOPBACK ON CH B /

016723      105     122     122 EM0017: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN EXTERNAL LOOP BACK ON CH A/

017020      105     122     122 EM0020: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN EXTERNAL LOOP BACK ON CH B/

017115      113     115     126 EM0023: .ASCIZ  /KMV11 REAL TIME CLOCK INTERRUPT OCCURED TOO EARLY /

017200      111     116     103 EM0024: .ASCIZ  /INCORRECT KMV11 REPLY /
```

GLOBAL ERROR REPORT SECTION

```
2330 017227      116      117      040  EM0027: .ASCIZ  /NO LOOP BACK CONNECTOR ,TEST NOT EXECUTED/
2331
2332 017301      104      101      124  EM0030: .ASCIZ  /DATA COMPARE ERROR BETWEEN MODEM SIGNAL 108 AND 107 ON CH A /
2333
2334 017376      104      101      124  EM0130: .ASCIZ  /DATA COMPARE ERROR BETWEEN MODEM SIGNAL 108 AND 107 ON CH B /
2335
2336 017473      115      117      104  EM0032: .ASCIZ  /MODEM SIGNAL ERROR ON CHANEL A IN EXTERNAL LOOPBACK MODE/
2337
2338 017564      115      117      104  EM0034: .ASCIZ  /MODEM SIGNAL ERROR ON CHANEL B IN EXTERNAL LOOPBACK MODE/
2339
2340 017655      120      122      117  EM0035: .ASCIZ  /PROM REVISION IS NOT COMPATIBLE WITH DIAGNOSTIC /
2341
2342
2343
2344
```

```
2346 017736      045     116     045  MLOOP:  .ASCIZ  /%N%A   NO LOOP BACK CONNECTOR,TEST NOT EXECUTED/
2347
2348 020014      045     116     045  MSEL0:  .ASCIZ  /%N%A SEL0 = %06%A   SHOULD BE = %06%N/
2349
2350 020062      045     116     045  MREG0:  .ASCIZ  /%N%A SEL0 = %06%A   SHOULD BE = %06/
2351 020126      045     116     045  MREG2:  .ASCIZ  /%N%A SEL2 = %06%A   SHOULD BE = %06/
2352 020172      045     116     045  MREG4:  .ASCIZ  /%N%A SEL4 = %06%A   SHOULD BE = %06/
2353 020236      045     116     045  MREG6:  .ASCIZ  /%N%A SEL6 = %06%A   SHOULD BE = %06/
2354 020302      045     116     045  MREG10: .ASCIZ  /%N%A SEL10= %06%A   SHOULD BE = %06/
2355 020346      045     116     045  MREG12: .ASCIZ  /%N%A SEL12= %06%A   SHOULD BE = %06/
2356 020412      045     116     045  MREG14: .ASCIZ  /%N%A SEL14= %06%A   SHOULD BE = %06/
2357 020456      045     116     045  MREG16: .ASCIZ  /%N%A SEL16= %06%A   SHOULD BE = %06/
2358
2359
2360
2361 020522      045     116     045  MINT:   .ASCIZ  /%N%A GOOD = %06%A  BAD =%06/
2362 020556      045     116     045  MBSEL0: .ASCIZ  /%N%A BSEL0 =%06%A  SHOULD BE =%06/
2363
2364 020620      045     116     045  MINTR:  .ASCIZ  /%N%A INTERRUPT OCCUR  WHEN ADDRESS =%06%A IS WRITTEN/
2365
2366 020705      045     116     045  MVECT:  .ASCIZ  /%N%A RECEIVE BAD VECT =%06%A SHOULD BE =%06/
2367
2368
2369
2370
2371
2372 020761      045     116     045  MT11V:  .ASCIZ  /%N%A   RECEIVE VECTOR VECT =%06%A SHOULD BE = %06/
2373 021043      045     116     045  MFRAM1: .ASCIZ  /%N%A   RECEIVE FRAME IS =%06%A SHOULD BE =%06/
2374 021121      045     116     045  MFRAM2: .ASCIZ  /%N%A   TRANSMIT SPEED IS  =%06%A FRAME LENGTH =%06/
2375
2376
2377 021204      045     116     045  MSTER1: .ASCIZ  /%N%A   ERROR STATUS  =%06/
2378 021236      045     116     045  MSTER2: .ASCIZ  /%N%A   WORD COUNT DISCREPANCY =%06/
2379
2380 021301      045     116     045  MODEM1: .ASCIZ  /%N%A   TESTED MODEM SIGNAL IS  =%06/
2381 021345      045     116     045  MODEM2: .ASCIZ  /%N%A   RESULT OF TEST IS  =%06/
2382 021404      045     116     045  MODEM3: .ASCIZ  /%N%A   MODEM SIGNAL STATE IS  =%06/
2383 021447      045     116     045  MRAMEF: .ASCIZ  /%N%A   TXDATA = %06%A      RXDATA = %06/
2384
```

```
                                        .EVEN

                                        ;-------------------------------------------------------------
                                        ; MACRO'S NEEDED TO REPORT ERRORS
                                        ;-------------------------------------------------------------

 021520                                         BGNMSG  PRSELO                          ;REPORT SELO
 021520                                         PRINTB  #MSELO,SELO,GOOD
 021550   004737  012740                        JSR     PC,CHKMAX
 021554                                         ENDMSG

 021556                                 BGNMSG  PRINT
 021556                                 PRINTB  #MINT,GOOD,BAD
 021606   004737  012740                JSR     PC,CHKMAX                               ;CHECK IF TOO MANY ERROR
 021612                                 ENDMSG

 021614                                 BGNMSG  PRALL                                   ;REPORT CONTENT OF ALL CSR'S
 021614                                 PRINTB  #MREG0,SEL0,GOOD0
 021644                                 PRINTB  #MREG2,SEL2,GOOD2
 021674                                 PRINTB  #MREG4,SEL4,GOOD4
 021724                                 PRINTB  #MREG6,SEL6,GOOD6
 021754                                 PRINTB  #MREG10,SEL10,GOOD10
 022004                                 PRINTB  #MREG12,SEL12,GOOD12
 022034                                 PRINTB  #MREG14,SEL14,GOOD14
 022064                                 PRINTB  #MREG16,SEL16,GOOD16
 022114   004737  012740                JSR     PC,CHKMAX                               ;CHECK IF TOO MANY ERROR
 022120                                 ENDMSG

 022122                                 BGNMSG  PRREG                                   ;REPORT ALL CSR'S BUT SELO
 022122                                 PRINTB  #MREG2,SEL2,GOOD2
 022152                                 PRINTB  #MREG4,SEL4,GOOD4
 022202                                 PRINTB  #MREG6,SEL6,GOOD6
 022232                                 PRINTB  #MREG10,SEL10,GOOD10
 022262                                 PRINTB  #MREG12,SEL12,GOOD12
 022312                                 PRINTB  #MREG14,SEL14,GOOD14
 022342                                 PRINTB  #MREG16,SEL16,GOOD16
 022372   004737  012740                JSR     PC,CHKMAX                               ;CHECK IF TOO MANY ERROR
 022376                                 ENDMSG

 022400                                 BGNMSG  PBSELO                                  ;REPORT BSELO
```

```
 2443 022400                          PRINTB  #MBSEL0,BSEL0,GOOD
 2444 022430  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2445 022434                          ENDMSG
 2446
 2447
 2448
 2449
 2450
 2451
 2452
 2453
 2454 022436                          BGNMSG  PINTR                        ;REPORT INTERRUPT
 2455 022436                          PRINTB  #MINTR,ADDR
 2456 022462  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2457 022466                          ENDMSG
 2458
 2459
 2460
 2461
 2462
 2463 022470                          BGNMSG  PVECT                        ;REPORT VECTOR
 2464 022470                          PRINTB  #MVECT,VECT,GOOD
 2465 022520  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2466 022524                          ENDMSG
 2467
 2468
 2469
 2470
 2471 022526                          BGNMSG  PRT11V
 2472 022526                          PRINTB  #MT11V,VECT,GOOD
 2473 022556  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2474 022562                          ENDMSG
 2475
 2476
 2477
 2478
 2479 022564                          BGNMSG  PFRAME                       ;REPORT FRAME ERROR
 2480 022564                          PRINTB  #MFRAM1,RXDATA,TXDATA
 2481 022614                          PRINTB  #MFRAM2,TSPEED,LENGTH
 2482 022644  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2483 022650                          ENDMSG
 2484
 2485
 2486
 2487
 2488
 2489 022652                          BGNMSG  PMODEM                       ;REPORT MODEM SIGNAL  ERROR
 2490 022652                          PRINTB  #MODEM1,GOOD
 2491 022676                          PRINTB  #MODEM2,BAD
 2492 022722                          PRINTB  #MODEM3,DATA
 2493 022746  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
 2494 022752                          ENDMSG
 2495
 2496
 2497
 2498
 2499
```

```
2500
2501
2502 022754                          BGNMSG  PRAMEF
2503 022754                          PRINTB  #MRAMEF,TXDATA,RXDATA          ;SHORT REPORT FOR FRAME ERROR
2504 023004                          ENDMSG
2505
2506
2507
2508
2509
2510
2511
2512
2513 023006                          BGNMSG  PRSTER                                 ;REPORT ERROR STATUS ,WORD CNT
2514 023006                          PRINTB  #MSTER1,STAERR
2515 023032                          PRINTB  #MSTER2,WRDCNT
2516 023056  004737  012740          JSR     PC,CHKMAX                      ;CHECK IF TOO MANY ERROR
2517 023062                          ENDMSG
2518
2519
2520
2521 023064                          BGNMSG  PADFLT                         ;ADDRESS TEST
2522 023064                          PRINTB  #TFM36,ADDR,UNIT
2523 023114  004737  012740          JSR     PC,CHKMAX
2524 023120                          ENDMSG
2525
2526
2527
2528
```

```
                                .SBTTL  REPORT CODING SECTION


                                ;++
                                ; THE REPORT CODING SECTION CONTAINS THE
                                ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
                                ;--

023122                                  BGNRPT


023122                                  EXIT    RPT


023126                                  ENDRPT
```

INITIALIZE SECTION

```
                                .SBTTL  INITIALIZE SECTION

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

023130                                  BGNINIT



                                        .EVEN

                                        .EVEN



                                ;INITIALIZE SUBROUTINE STACK
023130  012705  012676                  MOV     #SSTACK,R5
                                ;STORE BASE LEVEL PROGRAM STACK POINTER
023134  010637  002242                  MOV     SP,PSTACK
023140  005737  002252                  TST     FTIME
023144  001011                          BNE     1$
023146  013737  000004  002254          MOV     @#4,SAVE4
023154  013737  000006  002256          MOV     @#6,SAVE6
023162  012737  000001  002252          MOV     #1,FTIME
023170  013737  002254  000004  1$:     MOV     SAVE4,@#4
023176  013737  002256  000006          MOV     SAVE6,@#6

023204                                  READEF  #EF.START                       ;START COMMAND?
023212                                  BCOMPLETE       SETUP                   ;IF YES BRANCH

023214                                  READEF  #EF.CONTINUE                    ;CONTINUE COMMAND?
023222                                  BCOMPLETE       END

023224                                  READEF  #EF.NEW                         ;NEW PASS?
023232                                  BNCOMPLETE      NEXT                    ;IF NOT EXIT SETUP

023234  012737  177777  012436  SETUP:  MOV     #-1,UUT                         ;INITIALISE UNIT NUMBER

023242  005237  012436          NEXT:   INC     UUT                             ;POINT NEXT UNIT
023246  023737  012436  002262          CMP     UUT,L$UIT                       ;ALL DONE?
023254  001523                          BEQ     ABORT                           ;IF YES END OF PASS

023256  013737  012436  002264          MOV     UUT,UNIT
023264                                  PRINTF  #RUNNING,UNIT


023310                                  GPHARD  UUT,R1                          ;GET P TABLE
023320                                  BNCOMPLETE      NEXT                    ;IF NOT AVAILABLE GET NEXT


023322                          GETPRM:
                                                                ;GET ADDRESS OF KMV11
023322  011137  012454                  MOV     (R1),KMVCSR
                                                                ;GET POINTER TO KMV11 SEL02 REG
023326  011137  012456                  MOV     (R1),KMVP02
```

INITIALIZE SECTION

```
2650 023332  062737  000002  012456          ADD     #2,KMVP02
2651                                                                ;GET POINTER TO KMV11 PORT REG - SEL 4
2652 023340  011137  012460                  MOV     (R1),KMVP04
2653 023344  062737  000004  012460          ADD     #4,KMVP04
2654                                                                ;GET POINTER TO KMV11 PORT REG - SEL 6
2655 023352  011137  012462                  MOV     (R1),KMVP06
2656 023356  062737  000006  012462          ADD     #6,KMVP06
2657                                                                ;GET POINTER TO KMV11 REG 10
2658 023364  011137  012464                  MOV     (R1),KMVP10
2659 023370  062737  000010  012464          ADD     #10,KMVP10
2660                                                                ;GET POINTER TO KMV11 REG 12
2661 023376  011137  012466                  MOV     (R1),KMVP12
2662 023402  062737  000012  012466          ADD     #12,KMVP12
2663                                                                ;GET POINTER TO KMV11 REG 14
2664 023410  011137  012470                  MOV     (R1),KMVP14
2665 023414  062737  000014  012470          ADD     #14,KMVP14
2666                                                                ;GET POINTER TO KMV11 REG 16
2667 023422  012137  012472                  MOV     (R1)+,KMVP16
2668 023426  062737  000016  012472          ADD     #16,KMVP16
2669                                                                ;GET POINTER TO  VECTOR 0
2670 023434  011137  012440                  MOV     (R1),KMVV00
2671                                                                ;GET POINTER TO  VECTOR 2
2672 023440  011137  012446                  MOV     (R1),KMVV02
2673 023444  062737  000002  012446          ADD     #2,KMVV02
2674                                                                ;GET POINTER TO VECTOR 4
2675 023452  011137  012444                  MOV     (R1),KMVV04
2676 023456  062737  000004  012444          ADD     #4,KMVV04
2677                                                                ;GET POINTER TO VECTOR 6
2678 023464  012137  012450                  MOV     (R1)+,KMVV06
2679 023470  062737  000006  012450          ADD     #6,KMVV06
2680                                                                ;GET POINTER TO TX PRIORITY LEVEL
2681 023476  012137  012442                  MOV     (R1)+,KMVLVL
2682 023502  062737  000006  012452          ADD     #6,KMTLVL
2683                                                                ;GET LOOPBACK PARAMETERS:
2684 023510  011137  012474                  MOV     (R1),LOOP
2685
2686 023514  005037  002236                  CLR     ERRCNT          ;CLEAR ERROR COUNT
2687 023520                                  EXIT    INIT
2688
2689
2690
2691 023524                          ABORT:  DOCLN                           ;CLEAN UP AND ABORT PASS
2692 023526                                  EXIT INIT                       ;EXIT
2693
2694
2695 023532                          END:    ENDINIT
2696
2697
2698                                         .NLIST  BEX
2699 023534      045     116     045 RUNNING:        .ASCIZ  /%N%A RUNNING ON UNIT %D2%A  PASS TIME=3 MINUTES/
2700                                         .LIST   BEX
2701                                         .EVEN
2702
2703
2704
```

```
                                .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
023614                                  BGNAUTO


                                        ;DEVICE DOES NOT HAVE A 'READY"
023614  013701  012454          MOV     KMVCSR,R1       ;R1 CONTAINS BASE KMV11 ADDRESS
023620  012705  000007          MOV     #7,R5           ;7 REGISTERS TO BE TESTED
023624  012737  023656  000004  MOV     #2$,4           ;SET OUT TIMEOUT TRAP
023632  012737  000340  000006  MOV     #340,6          ;LEVEL 7
023640  005711          1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
023642  000240                  NOP
023644  062701  000002          ADD     #2,R1           ;NEXT REGISTER
023650  005305                  DEC     R5              ;DEC REGISTER COUNT
023652  001372                  BNE     1$              ;BR IF NOT LAST REGISTER
023654  000405                  BR      3$

023656  062706  000004  2$:     ADD     #4,SP
023662                          DODU    LOGDEV

023670  013737  002254  000004  3$:     MOV     SAVE4,4
023676  013737  002256  000006          MOV     SAVE6,6
023704                          ENDAUTO
```

```
                                .SBTTL  CLEANUP CODING SECTION

                                ;////////////////////////////////////////////////////////////////////
                                ;/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ;/ AT THE END OF EACH PASS.
                                ;////////////////////////////////////////////////////////////////////

023706                                  BGNCLN



023706                                  BRESET

023710                                  ENDCLN
```

DROP UNIT SECTION

```
                                .SBTTL  DROP UNIT SECTION

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

023712                                  BGNDU



                                        .EVEN

023712                                  PRINTF  #DROPD,R0       ;UNIT DROPPED

023734                                  EXIT    DU



                                        .NLIST  BEX
023740  045  116  045  DROPD:           .ASCIZ  /%N%A UNIT %D2%A DROPPED/
                                        .LIST   BEX
                                        .EVEN

023770                                  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.
                                ;////////////////////////////////////////////////////////////////////



023772                                  BGNAU
023772                                  ENDAU
```

```
                                .SBTTL  HARDWARE TESTS


                                ;START OF CODE BLOCK WHICH IS USED AS DATA
023774                          ROMMAP::++
                                ; TEST TO ...
                                ;--


                                ;       BGNTST


                                ;       EXIT    TST


                                        .EVEN
                                ;       ENDTST
```

```
2928 023774                                     BADHEAD
                                        ;********************** TEST1 **************
2929                                            ;*VERIFY THAT REFERENCING QBUS DEVICE REGISTERS
2930                                            ;*DOES NOT CAUSE TIME OUT TRAPS.
2931 023774                                     BADHEAD
                                        ;********************** TEST1 **************
2932
2933 023774                                     BGNTST
2934 023774  013701  012454                      MOV     KMVCSR,R1       ;R1 CONTAINS KMV11 ADDRESSES
2935 024000  012705  000007                      MOV     #7,R5           ;7 REGISTERS TO BE TESTED
2936 024004  012737  024042  000004              MOV     #2$,4           ;SET OUT TIMEOUT TRAP
2937 024012  012737  000340  000006              MOV     #340,6          ;LEVEL 7
2938 024020  005711                      1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
2939 024022  000240                              NOP
2940 024024                                      ESCAPE  TST
2941 024030  062701  000002                      ADD     #2,R1           ;NEXT REGISTER
2942 024034  005305                              DEC     R5              ;DEC REGISTER COUNT
2943 024036  001370                              BNE     1$              ;BR IF NOT LAST REGISTER
2944 024040  000415                              BR      3$
2945
2946 024042  062706  000004              2$:     ADD     #4,SP
2947 024046  010137  002362                      MOV     R1,ADDR
2948 024052  013737  012436  002264              MOV     UUT,UNIT
2949 024060                                      ERRHRD  0,TIM,PADFLT
2950 024070                                      ESCAPE  TST
2951
2952
2953 024074  013737  002254  000004      3$:     MOV     SAVE4,4
2954 024102  013737  002256  000006              MOV     SAVE6,6
2955 024110                                      ESCAPE  TST
2956
2957 024114                                     ENDTST
2958                                            .EVEN
2959
2960
```

```
2962 024116                          BADHEAD
                                     ;********************** TEST2 **************
2963                                 ;CHECK PROM REVISION TO SEE IF IT IS COMPATIBLE WITH DIAGNOSTIC
2964 024116                          BADHEAD
                                     ;********************** TEST2 **************
2965
2966
2967
2968
2969 024116                          STARS 1
2970                                 ;READ LOCATION 2 OF THE PROM (ADDRESS 160002) WHICH CONTAINS PROM VERSION
2971                                 ;       NUMBER
2972                                 ;CHECK IF DIAGNOSTIC AND PROM ARE COMPATIBLE AND GIVE AN ERROR IF NOT
2973 024116                          STARS 1
2974
2975
2976
2977
2978
2979 024116                          BGNTST
2980 024116  004737  014400                  JSR     PC,CLRKMV               ;CLEAR ALL REGISTERS
2981 024122  004737  014502                  JSR     PC,MAINM1               ;SET MAINT MODE
2982
2983
2984 024126  004537  014730          REVPRO: JSR     R5,READ                 ;READ LOCATION 160002
2985 024132  160002                          .WORD   160002
2986
2987
2988 024134  023737  012430  012374          CMP     GDREV,BAD               ;LOOK IF COMPATIBLE
2989 024142  001410                          BEQ     1$                      ;YES
2990
2991 024144                                  ERRHRD  7,EM0035                ;REPORT THE ERROR
2992 024154  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
2993 024160                                  ESCAPE TST
2994 024164                          1$:
2995 024164                          ENDTST
```

```
                                BADHEAD
                                ;********************** TEST3 **************
                                ;REAL TIME CLOCK TEST
                                BADHEAD
                                ;********************** TEST3 **************




                                STARS 1
                                ;TEST DESCRIPTION:
                                ;THIS TEST CHECK KMV11 REAL TIME CLOCK.
                                ;THE DCT11 FULLY EXECUTE THIS MICRO TEST AND GIVE A TEST RESULT
                                ;VIA CSR'S TO THE HOST COMPUTER.( TIMING IS CHECKED BY DCT11)
                                ;
                                ;
                                ;DCT11 ENABLE CLOCK,AND THEN SET UP CLOCK FOR 80 MS PERIOD
                                ;
                                ;
                                ;QBUS WAIT FOR AT LEAST 80 MS AND CHECK IF AN INTERRUPT OCCUR
                                ;ON DCT11 SHIP AT VECTOR 130
                                ;
                                ;TURN OF CLOCK, WAIT AGAIN FOR MORE THAN 80 MS AND CHECK THAT NO
                                ;INTERRUPT OCCUR
                                ;
                                ;ERROR REPORTING:       BSEL0=200       IF TIMEOUT DURING TEST
                                ;                       BSEL0=100       IF ERROR DURING TEST
                                ;                       BSEL0=TEST NUB  IF NO KMV11 ANSWER
                                ;                       BSEL0=0         IF TEST IS OK
                                ;
                                ;IF ERROR               SEL6=1          IF NO INTERRUPT OCCUR
                                ;                       SEL6=2          IF BAD VECTOR
                                ;                       SEL6=4          IF INTERRUPT OCCUR WHEN CLOCK
                                ;                                       IS NOT ENABLE
                                ;                       SEL6=10         INTERRUPT OCCUR TOO EARLY
                                ;
                                ;
                                ;
                                ;                       SEL2=EXPECTED VECTOR
                                ;                       SEL4=OBTAINED VECTOR
                                ;
                                ;
                                ; MICRO TEST NB= 27

                                STARS 1




                                BGNTST
```

```
3052 024166  004737  014400                    JSR     PC,CLRKMV           ;CLR REG
3053 024172  004737  014502                    JSR     PC,MAINM1           ;SET MAINT MODE
3054 024176  004537  014552          RTCLK:    JSR     R5,TSTNUB
3055 024202  000027                            .WORD   27
3056
3057 024204                                    WAITB   0,2                 ;WAIT FOR TEST EXECUTION
3058
3059
3060 024224  004737  013102                    JSR     PC,TSTERR           ;CHECK BSEL0
3061 024230  000520                            BR      1$                  ;TEST OK
3062 024232  000423                            BR      2$                  ;TIMEOUT ERROR
3063 024234  000432                            BR      3$                  ;NO KMV ANSWER
3064
3065
3066 024236  022777  000001  166216            CMP     #1,@KMVP06          ;ERROR DURING TEST ,SEE WHICH ONE
3067 024244  001436                            BEQ     4$                  ;NO INTERRUPT OCCUR
3068
3069 024246  022777  000002  166206            CMP     #2,@KMVP06
3070 024254  001442                            BEQ     5$                  ;INT ON BAD VECTOR
3071
3072 024256  022737  000004  012462            CMP     #4,KMVP06
3073 024264  001452                            BEQ     6$                  ;INT OCCUR WHEN CLOCK IS DISABLE
3074
3075
3076
3077 024266  022737  000010  012462            CMP     #10,KMVP06          ;INTERRUPT OCCUR TOO EARLY
3078 024274  001456                            BEQ     7$
3079
3080 024276  000137  024452                    JMP     10$                 ;WRONG KMV11 ANSWER
3081
3082
3083
3084
3085 024302                          2$:       ERRHRD  8,EM0006            ;TIMEOUT ERROR
3086 024312  004737  012740                    JSR     PC,CHKMAX           ;CHECK IF TOO MANY ERROR
3087 024316                                    ESCAPE  TST
3088
3089
3090
3091 024322                          3$:       ERRHRD  9,EM0004            ;NO KMV11 ANSWER
3092 024332  004737  012740                    JSR     PC,CHKMAX           ;CHECK IF TOO MANY ERROR
3093 024336                                    ESCAPE  TST
3094
3095
3096 024342                          4$:       ERRHRD  10,EM0011           ;NO INTERRUPT OCCUR
3097 024352  004737  012740                    JSR     PC,CHKMAX           ;CHECK IF TOO MANY ERROR
3098 024356                                    ESCAPE  TST
3099
3100
3101
3102 024362  017737  166072  012402    5$:     MOV     @KMVP04,VECT        ;READ BAD VECT
3103 024370  012737  000130  002272            MOV     #130,GOOD
3104 024376                                    ERRHRD  11,EM0007,PINTR     ;INTERRUPT OCCUR AT A BAD VECTOR
3105 024406                                    ESCAPE  TST
3106
3107
3108 024412                          6$:       ERRHRD  12,EM0012           ;INT OCCUR WHEN CHIP IS DISABLE
```

```
 3109 024422  004737  012740                JSR     PC,CHKMAX             ;CHECK IF TOO MANY ERROR
 3110 024426                                ESCAPE TST
 3111
 3112
 3113
 3114
 3115 024432                          7$:   ERRHRD  13,EM0023             ;INTERRUPT OCCURS EARLY
 3116 024442  004737  012740                JSR     PC,CHKMAX             ;CHECK IF TOO MANY ERROR
 3117 024446                                ESCAPE  TST
 3118
 3119
 3120
 3121
 3122
 3123
 3124 024452                          10$:  ERRHRD  14,EM0024             ;WRONG KMV11 ANSWER
 3125 024462  004737  012740                JSR     PC,CHKMAX             ;CHECK IF TOO MANY ERROR
 3126 024466                                ESCAPE  TST
 3127
 3128
 3129
 3130
 3131 024472  000240                   1$:  NOP
 3132 024474                                ENDTST
 3133
 3134
 3135
```

```
3137 024476                         BADHEAD
                                    ;********************** TEST4 **************
3138                                ;BAUD RATE GENERATOR TEST
3139 024476                         BADHEAD
                                    ;********************** TEST4 **************
3140
3141
3142
3143
3144
3145
3146
3147
3148 024476                         STARS 1
3149                                ;THIS TEST READ THE STATUS AND THE OUTPUT OF THE BAUD RATE GENERATOR
3150                                ;DURING EACH PHASE OF THE CLOCK PULSE
3151                                ;
3152                                ;
3153                                ;NOTE:THIS TEST AND ALL ITS VERIFICATIONS ARE MADE BY THE DCT11 WHICH
3154                                ;       ONLY GIVE TEST RESULT TO THE HOST VIA CSR'S
3155                                ;       ALL THE TIMING IS CHECKED BY THE DCT11
3156                                ;
3157                                ;
3158                                ;
3159                                ;TEST DESCRIPTION:
3160                                ;       -DCT11 LOAD GENE COUNT WITH MAX COUNT (=^4.74 MSEC)
3161                                ;
3162                                ;       -READ BACK GENE COUNT AND STATUS AND CHECK
3163
3164
3165
3166
3167                                ;STEP 1:READ COUNT AFTER STARTING CLOCK
3168                                ;       CLOCK COUNT MUST BE NEGATIVE
3169                                ;       OUTPUT MUST BE = 1
3170
3171
3172
3173                                ;        ERROR REPORTING:
3174                                ;        IF COUNT=POSITIVE      BSEL0=100=ERROR
3175                                ;                               SEL6 =1  =GENE COUNT CAN'T BE READ OR
3176                                ;                                         WRITE CORRECTLY
3177                                ;        IF OUTPUT-0            BSEL0=100=ERROR
3178                                ;                               SEL6 =2  =GENE OUTPUT ISN'T IN A GOOD
3179                                ;                                         STATE(NO ACTION)
3180
3181                                ;STEP 2:  WAIT 2,5MSEC AND READ BACK AGAIN GENERATOR COUNT AND STATUS
3182                                ;         OUTPUT MUST BE = 0
3183
3184
3185
3186
3187                                ;        ERROR REPORTING:
3188                                ;        IF OUTPUT =1           BSEL0=100= ERROR
3189                                ;                               SEL6=10  =OUTPUT ISN'T IN A GOOD STATE
3190
3191                                ;STEP3:WAIT 2.5 MSEC MORE AND READ BACK AGAIN GENERATOR COUNT AND STATUS
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 60-1
HARDWARE TESTS

                                ;               OUTPUT MUST BE = 1
                                ;
                                ;
                                ;
                                ;
                                ;       ERROR REPORTING:
                                ;
                                ;               IF OUTPUT=0             BSEL0=100=ERROR
                                ;                                       SEL6=40  =NO ACTION ON GENERATOR OUTPUT
                                ;       -ELSE EXIT
                                ;
                                ;
                                ;
                                ;
                                ;TEST 30=       TEST GENERATOR A
                                ;TEST 31=       TEST GENERATOR B
                                STARS 1
```

```
3210 024476                                BGNTST
3211 024476                                BGNSUB
3212 024500  004737  014400                       JSR     PC,CLRKMV       ;CLR REG
3213 024504  004737  014502                       JSR     PC,MAINM1       ;SET MAINT MODE
3214 024510  004537  014552                       JSR     R5,TSTNUB
3215 024514  000030                              .WORD   30
3216
3217
3218 024516                                BDRGEN: WAITB   0,1             ;WAIT FOR TEST EXECUTION
3219
3220 024536  004737  013102                       JSR     PC,TSTERR       ;CHECK BSEL0 TO SEE IF ERROR
3221 024542  000137  024742                       JMP     BDROK0          ;TEST OK BR AT END
3222 024546  000402                              BR      2$              ;TIME OUT ERROR
3223 024550  000401                              BR      2$              ;NO KMV11 ANSWER
3224 024552  000410                              BR      3$              ;ERROR DURING TEST
3225
3226
3227
3228 024554                                2$:     ERRHRD  15,EM0004       ;NO KMV11 ANSWER
3229 024564  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3230 024570                                       ESCAPE SUB
3231
3232
3233
3234 024574                                3$:                             ;LOOK WHICH ERROR
3235 024574  017737  165662  002326               MOV     @KMVP06,SEL6    ;READ SEL6
3236 024602  022737  000001  002326               CMP     #1,SEL6         ;LOOK IF ERROR 1
3237 024610  001010                              BNE     4$              ;NO
3238
3239 024612                                       ERRHRD  16,EM0012       ;GENE COUNT CAN'T BE READ OR WRITTEN CORRECTLY
3240 024622  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3241 024626                                       ESCAPE SUB
3242
3243
3244 024632  022737  000002  002326  4$:          CMP     #2,SEL6         ;LOOK IF ERROR 2
3245 024640  001010                              BNE     5$              ;NO
3246
3247
3248
3249 024642                                       ERRHRD  17,EM0013       ;GENE OUTPUT ISN'T IN A GOOD STATE
3250 024652  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3251 024656                                       ESCAPE SUB
3252
3253
3254
3255 024662  022737  000010  002326  5$:          CMP     #10,SEL6        ;IS IT ERROR10?
3256 024670  001414                              BEQ     GENOUT
3257 024672  022737  000040  002326               CMP     #40,SEL6
3258 024700  001410                              BEQ     GENOUT
3259
3260 024702                                       ERRHRD  18,EM0024       ;WRONG KMV11 ANSWER
3261 024712  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3262 024716                                       ESCAPE  SUB
3263
3264
3265
3266
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 61-1
HARDWARE TESTS

   3267
   3268
   3269 024722                          GENOUT: ERRHRD  19,EM0014       ;NO ACTION ON GENERATOR OUTPUT
   3270 024732  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
   3271 024736                                  ESCAPE SUB
   3272
   3273
   3274
   3275 024742                          BDROK0:
   3276 024742                          ENDSUB
```

```
3278
3279 024744                                BGNSUB
3280 024746  004737  014400                       JSR     PC,CLRKMV       ;CLR REG
3281 024752  004737  014502                       JSR     PC,MAINM1       ;SET MAINT MODE
3282 024756  004537  014552                       JSR     R5,TSTNUB
3283 024762  000031                              .WORD   31
3284
3285
3286 024764                                      WAITB   0,1             ;WAIT FOR TEST EXECUTION
3287
3288 025004  004737  013102                       JSR     PC,TSTERR       ;CHECK BSEL0 TO SEE IF ERROR
3289 025010  000137  025210                       JMP     BDROK1          ;TEST OK BR AT END
3290 025014  000402                              BR      2$              ;TIME OUT ERROR
3291 025016  000401                              BR      2$              ;NO KMV11 ANSWER
3292 025020  000410                              BR      3$              ;ERROR DURING TEST
3293
3294
3295
3296 025022                                2$:   ERRHRD  20,EM0004       ;NO KMV11 ANSWER
3297 025032  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3298 025036                                      ESCAPE SUB
3299
3300
3301
3302 025042                                3$:                           ;LOOK WHICH ERROR
3303 025042  017737  165414  002326               MOV     @KMVP06,SEL6    ;READ SEL6
3304 025050  022737  000001  002326               CMP     #1,SEL6         ;LOOK IF ERROR 1
3305 025056  001010                              BNE     4$              ;NO
3306
3307 025060                                      ERRHRD  21,EM0112       ;GENE COUNT CAN'T BE READ OR WRITE CORRECTLY
3308 025070  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3309 025074                                      ESCAPE SUB
3310
3311
3312 025100  022737  000002  002326  4$:         CMP     #2,SEL6         ;LOOK IF ERROR 2
3313 025106  001010                              BNE     5$              ;NO
3314
3315
3316
3317 025110                                      ERRHRD  22,EM0113       ;GENE OUTPUT ISN'T IN A GOOD STATE
3318 025120  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3319 025124                                      ESCAPE SUB
3320
3321
3322
3323 025130  022737  000010  002326  5$:         CMP     #10,SEL6        ;EROR10?
3324 025136  001414                              BEQ     GEN0
3325 025140  022737  000040  002326               CMP     #40,SEL6
3326 025146  001410                              BEQ     GEN0
3327
3328 025150                                      ERRHRD  23,EM0024       ;WRONG KMV11 ANSWER
3329 025160  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3330 025164                                      ESCAPE  SUB
3331
3332
3333
3334
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 62-1
HARDWARE TESTS

    3335
    3336
    3337 025170                          GENO:   ERRHRD  24,EM0014       ;NO ACTION ON GENERATOR OUTPUT
    3338 025200  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
    3339 025204                                  ESCAPE SUB
    3340
    3341
    3342
    3343 025210                          BDROK1:
    3344 025210                          ENDSUB
    3345 025212                          ENDTST
```

```
3347 025214                             BADHEAD
                                        ;********************** TEST5 **************
3348                                    ;TRANSMIT DIFFERENT FRAMES (OF 500 WORDS) IN INTERNAL
3349                                    ;MODE WITHOUT ANY INTERRUPT ON CHANNEL A
3350 025214                             BADHEAD
                                        ;********************** TEST5 **************
3351
3352
3353
3354
3355
3356
3357
3358 025214                             STARS 1
3359                                    ;QBUS WRITE DIFFERENT TX TABLE OF 500 WORDS, LOAD IN KMV11 CSR'S
3360                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
3361                                    ;
3362                                    ;
3363                                    ;
3364                                    ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA  AND WRITE BACK
3365                                    ;IN RX TABLE (TRANSFER FROM QBUS TO KMV11 =DMA)
3366                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
3367                                    ;RX TABLE =TX TABLE
3368                                    ;
3369                                    ;
3370                                    ;PARAMETERS SELECTION:
3371                                    ;       SEL2=   TX TABLE ADDRESS
3372                                    ;       SEL4=   TX TABLE LENGTH
3373                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
3374                                    ;       BSEL7=      ''          ''  RX   ''
3375                                    ;       SEL12=  RX TABLE ADDRESS
3376                                    ;       SEL14=  SPEED SELECTION
3377                                    ;       SEL16=  ERROR STATUS
3378                                    ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
3379                                    ;                                       >0 IF TX>RX,<0 IF TX<RX
3380                                    ;       BSEL0=  TEST STATUS
3381                                    ;
3382                                    ;
3383                                    ;
3384                                    ;TEST STATUS DESCRIPTION:
3385                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
3386                                    ;       BSEL0=  200     =TIMEOUT ERROR
3387                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
3388                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
3389                                    ;
3390                                    ;
3391                                    ;
3392                                    ;ERROR STATUS DESCRIPTION:
3393                                    ;
3394                                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
3395                                    ;
3396                                    ;
3397                                    ;       SEL16=  BIT14=1 =FCS ERROR
3398                                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
3399                                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
3400                                    ;       SEL16=  BIT7 =1 =RX ABORT ERROR
3401                                    ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
```

HARDWARE TESTS

```
                                ;       SEL16=  BIT5 =1 =BYTE COUNT DISCREPANCY
                                ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
                                ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
                                ;       SEL16=  BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
                                ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
                                ;                                                    ONLY DURING SELF TEST)
                                ;
                                ;MICRO DIAG TEST DESCRIPTION:
                                ;TEST 36        =TRANSMIT  FRAMES ON CHANNEL A WITHOUT INTERRUPT
                                ;
                                ;
025214                          STARS 1
```

```
3415 025214                              BGNTST
3416 025214  004737  014400                      JSR     PC,CLRKMV               ;CLR REG
3417 025220  005037  002250                      CLR     FLAG
3418 025224  004737  014502                      JSR     PC,MAINM1               ;SET MAINT MODE
3419 025230  012737  000500  012416              MOV     #500,LENGTH             ;SELECT LENGTH
3420
3421 025236  012737  000174  012414              MOV     #KB56,TSPEED            ;SELECT SPEED
3422
3423 025244  012703  000001          INTTX:      MOV     #1,R3                   ;SELECT A PATTERN
3424
3425
3426 025250  005203                  TXSTAR:     INC     R3                      ;NEW ONE
3427 025252  013704  012416                      MOV     LENGTH,R4               ;LOAD LENGTH
3428 025256  012702  002370                      MOV     #TTABLE,R2              ;TX TABLE ADDRESS
3429 025262  004737  013160          10$:        JSR     PC,GENER                ;WRITE TX TABLE
3430 025266  013722  012400                      MOV     DATA,(R2)+
3431 025272  005304                              DEC     R4                      ;ALL DONE?
3432 025274  001372                              BNE     10$
3433
3434
3435
3436 025276  013704  012416                      MOV     LENGTH,R4
3437 025302  012702  006370                      MOV     #RTABLE,R2              ;CLEAR RX TABLE
3438 025306  005022                  11$:        CLR     (R2)+
3439 025310  005304                              DEC     R4
3440 025312  001375                              BNE     11$
3441
3442
3443
3444
3445 025314  013777  012414  165146              MOV     TSPEED,@KMVP14          ;SEND TX SPEED
3446 025322  012777  002370  165126              MOV     #TTABLE,@KMVP02         ;SEND TX TABLE ADDRESS
3447 025330  013777  012416  165122              MOV     LENGTH,@KMVP04          ;LOAD TX TABLE ADDRESS
3448 025336  012777  006370  165122              MOV     #RTABLE,@KMVP12         ;LOAD RX TABLE ADDRESS
3449 025344  005077  165112                      CLR     @KMVP06
3450
3451
3452
3453
3454
3455
3456 025350  004537  014552                      JSR     R5,TSTNUB
3457 025354  000036                              .WORD   36                      ;DO TEST 36= CHA TEST
3458
3459
3460
3461 025356                          2$:         WAITB   0,2                     ;WAIT FOR TEST EXECUTION
3462
3463
3464 025376  004737  013102                      JSR     PC,TSTERR               ;CHECK BSEL0
3465
3466 025402  000427                              BR      6$                      ;TEST OK CHECK RX TABLE
3467 025404  000402                              BR      3$                      ;TIMEOUT ERROR
3468 025406  000401                              BR      3$                      ;NO KMV11 ANSWER
3469 025410  000410                              BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
3470
3471
```

```
3472 025412                          3$:     ERRHRD  25,EM0004              ;NO KMV11 ANSWER
3473 025422  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3474 025426                                  ESCAPE  TST
3475
3476
3477
3478 025432                          4$:                                     ;ERROR DURING TEST READ ERROR STATUS
3479                                                                         ;TO CHECK WHICH ONE
3480
3481 025432  017737  165034  012424          MOV     @KMVP16,STAERR          ;READ ERROR STATUS
3482
3483 025440  017737  165020  012426          MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
3484
3485 025446                                  ERRHRD  26,EM0015,PRSTER        ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3486                                                                         ;GIVE ERROR STATUS,WORD COUNT DISCREPANCY
3487 025456                                  ESCAPE  TST
3488
3489
3490
3491
3492
3493
3494
3495
3496 025462  012702  002370          6$:     MOV     #TTABLE,R2              ;LOAD TXTABLE ADDRESS
3497 025466  012705  006370                  MOV     #RTABLE,R5              ;   "  RXTABLE ADDRESS
3498 025472  013704  012416                  MOV     LENGTH,R4               ;TABLE LENGTH
3499
3500 025476  022225                  RXCK:   CMP     (R2)+,(R5)+             ;CHECK RX AND TX TABLE
3501 025500  001007                          BNE     RXERR
3502 025502  005304                          DEC     R4                      ;ALL CHECK?
3503 025504  001374                          BNE     RXCK                    ;NO BRANCH
3504
3505
3506
3507 025506  022703  000004                  CMP     #4,R3                   ;ALL KIND OF PATTERN DONE?
3508 025512  001256                          BNE     TXSTAR                  ;NO TRY WITH NEW ONE
3509
3510
3511
3512 025514  000137  025636                  JMP     RXEND
3513
3514 025520  162705  000002          RXERR:  SUB     #2,R5
3515 025524  162702  000002                  SUB     #2,R2
3516
3517 025530  011237  012410                  MOV     (R2),TXDATA
3518 025534  011537  012412                  MOV     (R5),RXDATA
3519
3520 025540  005737  002250                  TST     FLAG                    ;LOOK IF 1ST ERROR
3521 025544  001014                          BNE     7$
3522
3523 025546                                  ERRHRD  27,EM0015,PFRAME        ;DATA COMPARE ERROR
3524 025556  005237  002250                  INC     FLAG
3525 025562  062702  000002                  ADD     #2,R2                   ;POINT NEXT ADDRESS
3526 025566  062705  000002                  ADD     #2,R5
3527 025572  000137  025476                  JMP     RXCK
3528
```

```
 3529 025576                              7$:     ERRHRD  27,0,PRAMEF           ;SHORT REPORT
 3530 025606  005237  002250                      INC     FLAG
 3531 025612  062702  000002                      ADD     #2,R2
 3532 025616  062705  000002                      ADD     #2,R5                  ;POINT NEXT ADDRESS
 3533 025622  022737  000010  002250              CMP     #10,FLAG               ;LOOK IF 10 REPORT
 3534 025630  001322                              BNE     RXCK
 3535
 3536 025632                                      ESCAPE  TST
 3537
 3538
 3539 025636                              RXEND:
 3540
 3541
 3542
 3543 025636                              ENDTST
```

```
3545 025640                             BADHEAD
                                        ;********************** TEST6 **************
3546                                    ;TRANSMIT DIFFERENT FRAMES (OF 500 WORDS) IN INTERNAL
3547                                    ;MODE WITHOUT ANY INTERRUPT ON CHANNEL B
3548 025640                             BADHEAD
                                        ;********************** TEST6 **************
3549
3550
3551
3552
3553
3554
3555
3556 025640                             STARS 1
3557                                    ;QBUS WRITE DIFFERENT TX TABLE OF 500 WORDS, LOAD IN KMV11 CSR'S
3558                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
3559                                    ;
3560                                    ;
3561                                    ;
3562                                    ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHB AND WRITE BACK
3563                                    ;IN RX TABLE (TRANSFER FROM QBUS TO KMV11 =DMA)
3564                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
3565                                    ;RX TABLE =TX TABLE
3566                                    ;
3567                                    ;
3568                                    ;PARAMETERS SELECTION:
3569                                    ;       SEL2=   TX TABLE ADDRESS
3570                                    ;       SEL4=   TX TABLE LENGTH
3571                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
3572                                    ;       BSEL7=      ''          ''   RX   ''
3573                                    ;       SEL12=  RX TABLE ADDRESS
3574                                    ;       SEL14=  SPEED SELECTION
3575                                    ;       SEL16=  ERROR STATUS
3576                                    ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
3577                                    ;                                       >0 IF TX>RX,<0 IF TX<RX
3578                                    ;       BSEL0=  TEST STATUS
3579                                    ;
3580                                    ;
3581                                    ;
3582                                    ;TEST STATUS DESCRIPTION:
3583                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
3584                                    ;       BSEL0=  200     =TIMEOUT ERROR
3585                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
3586                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
3587                                    ;
3588                                    ;
3589                                    ;
3590                                    ;ERROR STATUS DESCRIPTION:
3591                                    ;
3592                                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
3593                                    ;
3594                                    ;
3595                                    ;       SEL16=  BIT14=1 =FCS ERROR
3596                                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
3597                                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
3598                                    ;       SEL16=  BIT7 =1 =RX ABORT ERROR
3599                                    ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
```

```
                                ;       SEL16=  BIT5 =1 =BYTE COUNT DISCREPANCY
                                ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
                                ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
                                ;       SEL16=  BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
                                ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
                                ;                                                    ONLY DURING SELF TEST)
                                ;
                                ;MICRO DIAG TEST DESCRIPTION:
                                ;TEST 37        =TRANSMIT  FRAMES ON CHANNEL B WITHOUT INTERRUPT
                                ;
                                ;
025640                          STARS 1
```

```
3613 025640                          BGNTST
3614 025640  004737  014400                  JSR     PC,CLRKMV               ;CLR REG
3615 025644  005037  002250                  CLR     FLAG
3616 025650  004737  014502                  JSR     PC,MAINM1               ;SET MAINT MODE
3617 025654  012737  000500  012416          MOV     #500,LENGTH             ;SELECT LENGTH
3618
3619 025662  012737  000174  012414          MOV     #KB56,TSPEED            ;SELECT SPEED
3620
3621 025670  012703  000001                  MOV     #1,R3                   ;SELECT A PATTERN
3622
3623
3624 025674  005203                BTXSTA:   INC     R3                      ;NEW ONE
3625 025676  013704  012416                  MOV     LENGTH,R4               ;LOAD LENGTH
3626 025702  012702  002370                  MOV     #TTABLE,R2              ;TX TABLE ADDRESS
3627 025706  004737  013160          10$:    JSR     PC,GENER                ;WRITE TX TABLE
3628 025712  013722  012400                  MOV     DATA,(R2)+
3629 025716  005304                          DEC     R4                      ;ALL DONE?
3630 025720  001372                          BNE     10$
3631
3632
3633
3634 025722  013704  012416                  MOV     LENGTH,R4
3635 025726  012702  006370                  MOV     #RTABLE,R2              ;CLEAR RX TABLE
3636 025732  005022                  11$:    CLR     (R2)+
3637 025734  005304                          DEC     R4
3638 025736  001375                          BNE     11$
3639
3640
3641
3642
3643 025740  013777  012414  164522          MOV     TSPEED,@KMVP14          ;SEND TX SPEED
3644 025746  012777  002370  164502          MOV     #TTABLE,@KMVP02         ;SEND TX TABLE ADDRESS
3645 025754  013777  012416  164476          MOV     LENGTH,@KMVP04          ;LOAD TX TABLE ADDRESS
3646 025762  012777  006370  164476          MOV     #RTABLE,@KMVP12         ;LOAD RX TABLE ADDRESS
3647 025770  005077  164466                  CLR     @KMVP06
3648
3649
3650
3651
3652
3653 025774  004537  014552          1$:     JSR     R5,TSTNUB
3654 026000  000037                          .WORD   37                      ;DO TEST 37= CHB TEST
3655
3656
3657
3658 026002                          2$:     WAITB   0,2                     ;WAIT FOR TEST EXECUTION
3659
3660
3661 026022  004737  013102                  JSR     PC,TSTERR               ;CHECK BSEL0
3662
3663 026026  000427                          BR      6$                      ;TEST OK CHECK RX TABLE
3664 026030  000402                          BR      3$                      ;TIMEOUT ERROR
3665 026032  000401                          BR      3$                      ;NO KMV11 ANSWER
3666 026034  000410                          BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
3667
3668
3669 026036                          3$:     ERRHRD  25,EM0004               ;NO KMV11 ANSWER
```

```
3670 026046  004737  012740                JSR     PC,CHKMAX           ;CHECK IF TOO MANY ERROR
3671 026052                                ESCAPE  TST
3672
3673
3674
3675 026056                          4$:                               ;ERROR DURING TEST READ ERROR STATUS
3676                                                                   ;TO CHECK WHICH ONE
3677
3678 026056  017737  164410  012424        MOV     @KMVP16,STAERR      ;READ ERROR STATUS
3679
3680 026064  017737  164374  012426        MOV     @KMVP10,WRDCNT      ;READ WORD COUNT DISCREPANCY
3681
3682 026072                                ERRHRD  26,EM0115,PRSTER    ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3683                                                                   ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3684 026102                                ESCAPE  TST
3685
3686
3687
3688
3689
3690 026106  012702  002370          6$:   MOV     #TTABLE,R2          :LOAD TXTABLE ADDRESS
3691 026112  012705  006370                MOV     #RTABLE,R5          ;   ''  RXTABLE ADDRESS
3692 026116  013704  012416                MOV     LENGTH,R4           ;TABLE LENGTH
3693
3694 026122  022225                  BRXCK:  CMP     (R2)+,(R5)+         ;CHECK RX AND TX TABLE
3695 026124  001007                        BNE     BRXERR
3696 026126  005304                        DEC     R4                  ;ALL CHECK?
3697 026130  001374                        BNE     BRXCK               ;NO BRANCH
3698
3699
3700
3701 026132  022703  000004                CMP     #4,R3               ;ALL KIND OF PATTERN DONE?
3702 026136  001256                        BNE     BTXSTA              ;NO TRY WITH NEW ONE
3703
3704
3705
3706 026140  000137  026262                JMP     BRXEND
3707
3708 026144  162705  000002          BRXERR: SUB     #2,R5
3709 026150  162702  000002                SUB     #2,R2
3710
3711 026154  011237  012410                MOV     (R2),TXDATA
3712 026160  011537  012412                MOV     (R5),RXDATA
3713
3714 026164  005737  002250                TST     FLAG                ;LOOK IF 1ST ERROR
3715 026170  001014                        BNE     7$
3716
3717 026172                                ERRHRD  27,EM0115,PFRAME    ;DATA CMP ERROR
3718 026202  005237  002250                INC     FLAG
3719 026206  062702  000002                ADD     #2,R2               ;POINT NEXT ADDRESS
3720 026212  062705  000002                ADD     #2,R5
3721 026216  000137  025476                JMP     RXCK
3722
3723 026222                          7$:   ERRHRD  27,0,PRAMEF         ;SHORT REPORT
3724 026232  005237  002250                INC     FLAG
3725 026236  062702  000002                ADD     #2,R2
3726 026242  062705  000002                ADD     #2,R5               ;POINT NEXT ADDRESS
```

```
 3727 026246  022737  000010  002250          CMP     #10,FLAG                ;LOOK IF 10 REPORT
 3728 026254  001322                          BNE     BRXCK
 3729
 3730 026256                                  ESCAPE  TST
 3731
 3732
 3733 026262                          BRXEND:
 3734
 3735
 3736
 3737 026262                          ENDTST
```

```
                                BADHEAD
                                ;********************** TEST7 **************
                                ;TRANSMIT DIFFERENT  FRAME OF VARIOUS LENGTH (FROM 2BYTES TO 2K BYTES)
                                ;AT 56KBAUDS IN INTERNAL  MODE ON CHANNEL A  (TRANSMISSION WITH INTERRUPT)
                                BADHEAD
                                ;********************** TEST7 **************

                                STARS 1
                                ;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
                                ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
                                ;
                                ;
                                ;
                                ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA AND WRITE BACK
                                ;IN RX TABLE
                                ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
                                ;RX TABLE =TX TABLE
                                ;SPEED=56KBAUDS
                                ;
                                ;
                                ;PARAMETERS SELECTION:
                                ;       SEL2=   TX TABLE ADDRESS
                                ;       SEL4=   TX TABLE LENGTH
                                ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
                                ;       BSEL7=      ''          ''      RX   ''
                                ;       SEL12=  RX TABLE ADDRESS
                                ;       SEL14=  SPEED SELECTION
                                ;       SEL16=  ERROR STATUS
                                ;       BSEL0=  TEST STATUS
                                ;
                                ;
                                ;TEST STATUS DESCRIPTION:
                                ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
                                ;       BSEL0=  200     =TIMEOUT ERROR
                                ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
                                ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
                                ;
                                ;
                                ;ERROR STATUS DESCRIPTION:
                                ;
                                ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
                                ;
                                ;
                                ;       SEL16=  BIT14=1 =FCS ERROR
                                ;       SEL16=  BIT13=1 =OVERRUN ERROR
                                ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
```

```
                                ;       SEL16=  BIT7 =1 =RX ABORT ERROR
                                ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
                                ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
                                ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
                                ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
                                ;       SEL16=  BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
                                ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
                                ;                                                   ONLY DURING SELF TEST)
                                ;
                                ;
                                ;MICRO DIAG TEST DESCRIPTION:
                                ;TEST 40        =TRANSMIT VARIOUS LENGTH FRAME  AT 56 KBAUDS ON CHANNEL A
                                ;
                                ;
026264                          STARS 1
```

```
                                        BGNTST
3811 026264  004737  014400                    JSR     PC,CLRKMV               ;CLR REG
3812 026270  004737  014502                    JSR     PC,MAINM1               ;SET MAINT MODE
3813 026274  005037  002250                    CLR     FLAG
3814
3815
3816 026300  012703  000004                    MOV     #4,R3                   ;SELECT RANDOM PATTERN
3817 026304  012737  000174  012414            MOV     #KB56,TSPEED            ;SELECT SPEED
3818
3819 026312  012737  000001  012416  TXLTAR:   MOV     #1,LENGTH               ;START WITH 2 CHARACTERS
3820
3821 026320  013704  012416          TXLBGN:   MOV     LENGTH,R4
3822 026324  012702  002370                    MOV     #TTABLE,R2
3823 026330                                    BREAK
3824 026332  004737  013160          10$:      JSR     PC,GENER                ;WRITE TX TABLE
3825 026336  013722  012400                    MOV     DATA,(R2)+
3826 026342  005304                            DEC     R4
3827 026344  001372                            BNE     10$
3828
3829
3830 026346  013704  012416                    MOV     LENGTH,R4               ;CLEAR RX TABLE
3831 026352  012702  006370                    MOV     #RTABLE,R2
3832 026356  005022                  20$:      CLR     (R2)+
3833 026360  005304                            DEC     R4
3834 026362  001375                            BNE     20$
3835
3836
3837
3838
3839
3840
3841 026364  013777  012414  164076            MOV     TSPEED,@KMVP14          ;SEND TX SPEED
3842 026372  012777  002370  164056            MOV     #TTABLE,@KMVP02         ;  "  TX TABLE ADDRESS
3843 026400  013777  012416  164052            MOV     LENGTH,@KMVP04          ;  "  "     "    LENGTH
3844 026406  012777  006370  164052            MOV     #RTABLE,@KMVP12         ;SEND RX TABLE ADDRESS
3845 026414  005077  164042                    CLR     @KMVP06                 ;CLR EXTENDED ADDRESS
3846
3847
3848
3849
3850 026420  004537  014552                    JSR     R5,TSTNUB
3851 026424  000040                            .WORD   40                      ;DO TEST 40= CHA TEST
3852
3853
3854
3855 026426                          2$:       WAITB   0,2                     ;WAIT FOR TEST EXECUTION
3856
3857
3858 026446  004737  013102                    JSR     PC,TSTERR               ;CHECK BSEL0
3859
3860 026452  000427                            BR      6$                      ;TEST OK CHECK RX TABLE
3861 026454  000402                            BR      3$                      ;TIMEOUT ERROR
3862 026456  000401                            BR      3$                      ;NO KMV11 ANSWER
3863 026460  000410                            BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
3864
3865
3866 026462                          3$:       ERRHRD  28,EM0004               ;NO KMV11 ANSWER
```

```
3867 026472  004737  012740              JSR     PC,CHKMAX           ;CHECK IF TOO MANY ERROR
3868 026476                              ESCAPE  TST
3869
3870 026502                      4$:                                 ;ERROR DURING TEST READ ERROR STATUS
3871                                                                 ;TO CHECK WHICH ONE
3872
3873 026502  017737  163764  012424      MOV     @KMVP16,STAERR      ;READ ERROR STATUS
3874
3875 026510  017737  163750  012426      MOV     @KMVP10,WRDCNT      ;READ WORD COUNT DISCREPANCY
3876
3877 026516                              ERRHRD  29,EM0016,PRSTER    ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3878                                                                 ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3879 026526                              ESCAPE  TST
3880
3881
3882
3883
3884 026532  012702  002370      6$:     MOV     #TTABLE,R2          ;LOAD TX TABLE ADDRESS
3885 026536  012705  006370              MOV     #RTABLE,R5          ; ''   RX   ''     ''
3886 026542  013704  012416              MOV     LENGTH,R4           ; ''   TX TABLE LENGTH
3887
3888
3889 026546  022522              RXLCK:  CMP     (R5)+,(R2)+         ;CMP TX AND RX TABLE
3890 026550  001015                      BNE     RXLERR              ;BR IF ERROR
3891 026552  005304                      DEC     R4                  ;ALL DONE
3892 026554  001374                      BNE     RXLCK               ;NO
3893
3894 026556  062737  000400  012416      ADD     #400,LENGTH         ;CHANGE LENGTH
3895 026564  022737  002000  012416      CMP     #2000,LENGTH        ;IS IT MAX?
3896 026572  100252                      BPL     TXLBGN              ;NO DO TEST AGAIN WITH NEW TABLE
3897                                                                 ;                       LENGTH
3898
3899 026574  005303                      DEC     R3                  ;SELECT OTHER PATERNS
3900 026576  001245                      BNE     TXLTAR
3901
3902 026600  000137  026722              JMP     RXLEND
3903
3904
3905
3906 026604  162705  000002      RXLERR: SUB     #2,R5
3907 026610  162702  000002              SUB     #2,R2
3908
3909 026614  011237  012410              MOV     (R2),TXDATA
3910 026620  011537  012412              MOV     (R5),RXDATA
3911
3912 026624  005737  002250              TST     FLAG                ;LOOK IF 1ST ERROR
3913 026630  001014                      BNE     30$
3914
3915 026632                              ERRHRD  30,EM0016,PFRAME    ;DATA CMP ERROR
3916 026642  005237  002250              INC     FLAG
3917 026646  062702  000002              ADD     #2,R2               ;POINT NEXT ADDRESS
3918 026652  062705  000002              ADD     #2,R5
3919 026656  000137  025476              JMP     RXCK
3920
3921 026662                      30$:    ERRHRD  30,0,PRAMEF         ;SHORT REPORT
3922 026672  005237  002250              INC     FLAG
3923 026676  062702  000002              ADD     #2,R2
```

```
   3924 026702  062705  000002                ADD     #2,R5                   ;POINT NEXT ADDRESS
   3925 026706  022737  000010  002250        CMP     #10,FLAG                ;LOOK IF 10 REPORT
   3926 026714  001314                        BNE     RXLCK
   3927
   3928 026716                                ESCAPE  TST
   3929
   3930
   3931
   3932
   3933 026722                        RXLEND:
   3934 026722                        ENDTST
```

```
                                BADHEAD
                                ;********************** TEST8 **************
                                ;TRANSMIT DIFFERENT  FRAME OF VARIOUS LENGTH (FROM 2BYTES TO 2K BYTES)
                                ;AT 56KBAUDS IN INTERNAL  MODE ON CHANNEL  B (TRANSMISSION WITH INTERRUPT)
                                BADHEAD
                                ;********************** TEST8 **************

                                STARS 1
                                ;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
                                ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
                                ;
                                ;
                                ;
                                ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA AND CHB AND WRITE BACK
                                ;IN RX TABLE
                                ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
                                ;RX TABLE =TX TABLE
                                ;SPEED=56KBAUDS
                                ;
                                ;
                                ;PARAMETERS SELECTION:
                                ;       SEL2=   TX TABLE ADDRESS
                                ;       SEL4=   TX TABLE LENGTH
                                ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
                                ;       BSEL7=          ''              RX   ''
                                ;       SEL12=  RX TABLE ADDRESS
                                ;       SEL14=  SPEED SELECTION
                                ;       SEL16=  ERROR STATUS
                                ;       BSEL0=  TEST STATUS
                                ;
                                ;
                                ;TEST STATUS DESCRIPTION:
                                ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
                                ;       BSEL0=  200     =TIMEOUT ERROR
                                ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
                                ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
                                ;
                                ;
                                ;ERROR STATUS DESCRIPTION:
                                ;
                                ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
                                ;
                                ;
                                ;       SEL16=  BIT14=1 =FCS ERROR
                                ;       SEL16=  BIT13=1 =OVERRUN ERROR
                                ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
```

```
                                ;       SEL16=  BIT7 =1 =RX ABORT ERROR
                                ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
                                ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
                                ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
                                ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
                                ;       SEL16=  BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
                                ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USED
                                ;                                                    ONLY DURING SELF TEST)
                                ;
                                ;
                                ;MICRO DIAG TEST DESCRIPTION:
                                ;TEST 41         =TRANSMIT VARIOUS LENGTH FRAME  AT 56 KBAUDS ON CHANNEL B
                                ;
                                ;
 026724                         STARS 1
```

```
                                        BGNTST
4008 026724
4009 026724  004737  014400                     JSR     PC,CLRKMV               ;CLR REG
4010 026730  004737  014502                     JSR     PC,MAINM1               ;SET MAINT MODE
4011 026734  005037  002250                     CLR     FLAG
4012
4013
4014 026740  012703  000004                     MOV     #4,R3                   ;SELECT RANDOM PATTERN
4015 026744  012737  000174  012414             MOV     #KB56,TSPEED            ;SELECT SPEED
4016
4017 026752  012737  000001  012416 BXLTAR:    MOV     #1,LENGTH               ;START WITH 2 CHARACTERS
4018
4019 026760  013704  012416        BXLBGN:    MOV     LENGTH,R4
4020 026764  012702  002370                     MOV     #TTABLE,R2
4021 026770                                     BREAK
4022 026772  004737  013160        10$:       JSR     PC,GENER                ;WRITE TX TABLE
4023 026776  013722  012400                     MOV     DATA,(R2)+
4024 027002  005304                             DEC     R4
4025 027004  001372                             BNE     10$
4026
4027
4028 027006  013704  012416                     MOV     LENGTH,R4               ;CLEAR RX TABLE
4029 027012  012702  006370                     MOV     #RTABLE,R2
4030 027016  005022                20$:        CLR     (R2)+
4031 027020  005304                             DEC     R4
4032 027022  001375                             BNE     20$
4033
4034
4035
4036
4037
4038
4039 027024  013777  012414  163436             MOV     TSPEED,@KMVP14          ;SEND TX SPEED
4040 027032  012777  002370  163416             MOV     #TTABLE,@KMVP02         ;   "  TX TABLE ADDRESS
4041 027040  013777  012416  163412             MOV     LENGTH,@KMVP04          ;   "  "     "     LENGTH
4042 027046  012777  006370  163412             MOV     #RTABLE,@KMVP12         ;SEND RX TABLE ADDRESS
4043 027054  0050?   163402                     CLR     @KMVP06                 ;CLR EXTENDED ADDRESS
4044
4045
4046
4047 027060  004537  014552                     JSR     R5,TSTNUB
4048 027064  000041                             .WORD   41                      ;DO TEST 41= CHB TEST
4049
4050
4051 027066                        2$:         WAITB   0,2                     ;WAIT FOR TEST EXECUTION
4052
4053
4054 027106  004737  013102                     JSR     PC,TSTERR               ;CHECK BSEL0
4055
4056 027112  000427                             BR      6$                      ;TEST OK CHECK RX TABLE
4057 027114  000402                             BR      3$                      ;TIMEOUT ERROR
4058 027116  000401                             BR      3$                      ;NO KMV11 ANSWER
4059 027120  000410                             BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
4060
4061
4062 027122                        3$:         ERRHRD  28,EM0004               ;NO KMV11 ANSWER
4063 027132  004737  012740                     JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
4064 027136                                     ESCAPE  TST
```

```
4065
4066 027142                          4$:                                    ;ERROR DURING TEST READ ERROR STATUS
4067                                                                        ;TO CHECK WHICH ONE
4068
4069 027142  017737  163324  012424          MOV     @KMVP16,STAERR         ;READ ERROR STATUS
4070
4071 027150  017737  163310  012426          MOV     @KMVP10,WRDCNT         ;READ WORD COUNT DISCREPANCY
4072
4073 027156                                  ERRHRD  29,EM0116,PRSTER       ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
4074                                                                        ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
4075 027166                                  ESCAPE  TST
4076
4077
4078
4079
4080 027172  012702  002370          6$:     MOV     #TTABLE,R2             ;LOAD TX TABLE ADDRESS
4081 027176  012705  006370                  MOV     #RTABLE,R5             ; ''   RX    ''
4082 027202  013704  012416                  MOV     LENGTH,R4              ; ''   TX TABLE LENGTH
4083
4084
4085 027206  022522                  BXLCK:  CMP     (R5)+,(R2)+            ;CMP TX AND RX TABLE
4086 027210  001015                          BNE     BXLERR                 ;BR IF ERROR
4087 027212  005304                          DEC     R4                     ;ALL DONE
4088 027214  001374                          BNE     BXLCK                  ;NO
4089
4090 027216  062737  000400  012416          ADD     #400,LENGTH            ;CHANGE LENGTH
4091 027224  022737  002000  012416          CMP     #2000,LENGTH           ;IS IT MAX?
4092 027232  100252                          BPL     BXLBGN                 ;NO DO TEST AGAIN WITH NEW TABLE
4093                                                                        ;                        LENGTH
4094
4095 027234  005303                          DEC     R3                     ;SELECT OTHER PATERNS
4096 027236  001245                          BNE     BXLTAR
4097
4098 027240  000137  027362                  JMP     BXLEND
4099
4100
4101
4102 027244  162705  000002          BXLERR: SUB     #2,R5
4103 027250  162702  000002                  SUB     #2,R2
4104
4105 027254  011237  012410                  MOV     (R2),TXDATA
4106 027260  011537  012412                  MOV     (R5),RXDATA
4107
4108 027264  005737  002250                  TST     FLAG                   ;LOOK IF 1ST ERROR
4109 027270  001014                          BNE     30$
4110
4111 027272                                  ERRHRD  30,EM0116,PFRAME       ;DATA CMP ERROR
4112 027302  005237  002250                  INC     FLAG
4113 027306  062702  000002                  ADD     #2,R2                  ;POINT NEXT ADDRESS
4114 027312  062705  000002                  ADD     #2,R5
4115 027316  000137  000000G                 JMP     BXCK
4116
4117 027322                          30$:    ERRHRD  30,0,PRAMEF            ;SHORT REPORT
4118 027332  005237  002250                  INC     FLAG
4119 027336  062702  000002                  ADD     #2,R2
4120 027342  062705  000002                  ADD     #2,R5                  ;POINT NEXT ADDRESS
4121 027346  022737  000010  002250          CMP     #10,FLAG               ;LOOK IF 10 REPORT
```

```
   4122 027354  001314                          BNE     BXLCK
   4123
   4124 027356                                  ESCAPE  ;ST
   4125
   4126
   4127
   4128
   4129 027362                          BXLEND:
   4130 027362                          ENDTST
```

```
4132
4133 027364                             BADHEAD
                                        ;************************ TEST9 ***************
4134                                    ;TRANSMIT DIFFERENT FRAMES OF VARIOUS LENGTH IN EXTERNAL LOOP BACK
4135                                    ;MODE ON CHANNEL A
4136 027364                             BADHEAD
                                        ;************************ TEST9 ***************
4137
4138
4139
4140
4141
4142
4143 027364                             STARS 1
4144                                    ;
4145                                    ;AT BEGINNING OF TEST ,CHECK IF LOOP BACK CONNECTORS ARE INSTALLED
4146                                    ;OR NOT:IF NOT INSTALLED = EXIT TEST AND GIVE ERROR MESSAGE
4147                                    ;                          ************************************
4148                                    ;
4149                                    ;
4150                                    ;
4151                                    ;
4152                                    ;
4153                                    ;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
4154                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED (56KB)
4155                                    ;
4156                                    ;
4157                                    ;
4158                                    ;DCT11 EXECUTE THE TRANSFER IN EXTERNAL MODE ON CHA  AND WRITE BACK
4159                                    ;IN RX TABLE
4160                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
4161                                    ;RX TABLE =TX TABLE
4162                                    ;
4163                                    ;
4164                                    ;PARAMETERS SELECTION:
4165                                    ;       SEL2=   TX TABLE ADDRESS
4166                                    ;       SEL4=   TX TABLE LENGTH
4167                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
4168                                    ;       BSEL7=      ''          ''    RX   ''
4169                                    ;       SEL12=  RX TABLE ADDRESS
4170                                    ;       SEL14=  SPEED SELECTION
4171                                    ;       SEL16=  ERROR STATUS
4172                                    ;       BSEL0=  TEST STATUS
4173                                    ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
4174                                    ;                                       >0 IF TX>RX,<0 IF TX<RX
4175                                    ;
4176                                    ;
4177                                    ;
4178                                    ;TEST STATUS DESCRIPTION:
4179                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
4180                                    ;       BSEL0=  200     =TIMEOUT ERROR
4181                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
4182                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
4183                                    ;
4184                                    ;
4185                                    ;
4186                                    ;ERROR STATUS DESCRIPTION:
```

```
4187
4188                            ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
4189                            ;
4190                            ;
4191                            ;       SEL16=  BIT14=1 =FCS ERROR
4192                            ;       SEL16=  BIT13=1 =OVERRUN ERROR
4193                            ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
4194                            ;       SEL16=  BIT7 =1 =RX ABORT ERROR
4195                            ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
4196                            ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
4197                            ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
4198                            ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
4199                            ;       SEL16= BIT2 =1 =CLOCK PROBLEM
4200                            ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
4201                            ;                                                   ONLY DURING SELF TEST)
4202                            ;
4203                            ;
4204                            ;MICRO DIAG TEST DESCRIPTION:
4205                            ;TEST 42        =TRANSMIT VARIOUS LENGTH FRAME AT 56 KBAUDS SPEED ON CHANNEL A
4206                            ;                       IN EXTERNAL LOOP BACK MODE
4207                            ;
4208                            ;
4209                            ;
4210                            ;
4211                            ;CAUTION:
4212                            ;--------
4213                            ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4214                            ;
4215                            ;
4216                            ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4217                            ;
4218                            ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4219                            ;EXTERNAL LOOP BACK CONECTOR
4220                            ;
4221                            ;
4222                            ;
4223                            ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4224                            ;MODEM CABLE.
4225                            ;
4226                            ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
4227                            ;MODEM CABLE.
4228                            ;
4229 027364                    STARS 1
```

```
4231 027364                              BGNTST
4232 027364  004737  014400                      JSR     PC,CLRKMV          ;CLEAR REGISTERS
4233 027370  0C4737  014600                      JSR     PC,CKKMV           ;LOOK IF LOOP BACK CON INSTALLED?
4234 027374  005737  012474                      TST     LOOP               ;IS LOOP BIT=1?
4235 027400  001412                             BEQ     BGNTXA             ;YES GO ON TEST
4236 027402                                      PRINTF  #MLOOP             ;NO LOOP BACK PLUGGED .THE MODULE
4237                                                                        ;WILL NOT BE TESTED IS EXTERNAL LOOP
4238 027422  000137  030060                      JMP     RXAEND
4239
4240
4241
4242 027426  004737  014502              BGNTXA: JSR     PC,MAINM1          ;SET MAINT MODE
4243 027432  005037  002250                      CLR     FLAG
4244
4245 027436  012703  000004                      MOV     #4,R3              ;SELECT RANDOM PATTERN
4246 027442  012737  000174  012414              MOV     #KB56,TSPEED       ;SELECT SPEED
4247
4248 027450  012737  000001  012416      TXATAR: MOV     #1,LENGTH          ;1ST TABLE LENGTH(200 WORDS)
4249
4250 027456  013704  012416              TXABGN: MOV     LENGTH,R4
4251 027462  012702  002370                      MOV     #TTABLE,R2
4252 027466                                      BREAK
4253 027470  004737  013160              10$:    JSR     PC,GENER           ;WRITE TABLE
4254 027474  013722  012400                      MOV     DATA,(R2)+
4255 027500  005304                             DEC     R4
4256 027502  001372                             BNE     10$
4257
4258
4259
4260 027504  013704  012416                      MOV     LENGTH,R4          ;CLEAR RX TABLE
4261 027510  012702  006370                      MOV     #RTABLE,R2
4262 027514  005022                      20$:    CLR     (R2)+
4263 027516  005304                             DEC     R4
4264 027520  001375                             BNE     20$
4265
4266
4267
4268
4269
4270
4271
4272 027522  013777  012414  162740              MOV     TSPEED,@KMVP14     ;SEND TX SPEED
4273 027530  012777  002370  162720              MOV     #TTABLE,@KMVP02    ;  "  TX TABLE ADDRESS
4274 027536  013777  012416  162714              MOV     LENGTH,@KMVP04     ;  "  "     "    LENGTH
4275 027544  012777  006370  162714              MOV     #RTABLE,@KMVP12    ;SEND RX TABLE ADDRESS
4276 027552  005077  162704                      CLR     @KMVP06            ;CLR EXTENDED ADDRESS
4277
4278
4279
4280
4281
4282 027556  004537  014552              1$:     JSR     R5,TSTNUB
4283 027562  000042                             .WORD   42                 ;DO TEST 42= CHB TEST
4284
4285
4286
4287 027564                              2$:     WAITB   0,3                ;WAIT FOR TEST EXECUTION
```

```
4288
4289
4290 027604  004737  013102                  JSR     PC,TSTERR               ;CHECK BSEL0
4291
4292 027610  000427                          BR      6$                      ;TEST OK CHECK RX TABLE
4293 027612  000402                          BR      3$                      ;TIMEOUT ERROR
4294 027614  000401                          BR      3$                      ;NO KMV11 ANSWER
4295 027616  000410                          BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
4296
4297
4298 027620                          3$:     ERRHRD  32,EM0004               ;NO KMV11 ANSWER
4299 027630  004737  012740                  JSR     PC,CHKMAX               ;TOO MANY ERROR?
4300 027634                                  ESCAPE  TST
4301
4302
4303 027640                          4$:                                     ;ERROR DURING TEST READ ERROR STATUS
4304                                                                         ;TO CHECK WHICH ONE
4305
4306 027640  017737  162626  012424          MOV     @KMVP16,STAERR          ;READ ERROR STATUS
4307
4308 027646  017737  162612  012426          MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
4309
4310 027654                                  ERRHRD  33,EM0017,PRSTER                ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
4311                                                                         ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
4312 027664                                  ESCAPE  TST
4313
4314
4315
4316
4317
4318 027670  012702  002370          6$:     MOV     #TTABLE,R2              ;LOAD TABLE PARAMETERS
4319 027674  012705  006370                  MOV     #RTABLE,R5
4320 027700  013704  012416                  MOV     LENGTH,R4
4321
4322 027704  022225                  RXACK:  CMP     (R2)+,(R5)+             ;CHECK TX AND RX TABLE
4323 027706  001015                          BNE     RXAERR
4324 027710  005304                          DEC     R4
4325 027712  001374                          BNE     RXACK
4326
4327 027714  062737  000400  012416          ADD     #400,LENGTH             ;CHANGE LENGTH
4328 027722  022737  002000  012416          CMP     #2000,LENGTH
4329 027730  100252                          BPL     TXABGN
4330 027732  005303                          DEC     R3                      ;SELECT NEW PATERN
4331 027734  001245                          BNE     TXATAR                  ;ALL DONE
4332 027736  000137  030060                  JMP     RXAEND
4333
4334
4335
4336 027742  162705  000002          RXAERR: SUB     #2,R5
4337 027746  162702  000002                  SUB     #2,R2
4338
4339 027752  011237  012410                  MOV     (R2),TXDATA
4340 027756  011537  012412                  MOV     (R5),RXDATA
4341
4342 027762  005737  002250                  TST     FLAG                    ;LOOK IF 1ST ERROR
4343 027766  001014                          BNE     30$
4344
```

```
4345 027770                                    ERRHRD  34,EM0017,PFRAME        ;DATA CMP ERROR
4346 030000  005237  002250                    INC     FLAG
4347 030004  062702  000002                    ADD     #2,R2                   ;POINT NEXT ADDRESS
4348 030010  062705  000002                    ADD     #2,R5
4349 030014  000137  027704                    JMP     RXACK
4350
4351 030020                            30$:    ERRHRD  34,0,PRAMEF             ;SHORT REPORT
4352 030030  005237  002250                    INC     FLAG
4353 030034  062702  000002                    ADD     #2,R2
4354 030040  062705  000002                    ADD     #2,R5                   ;POINT NEXT ADDRESS
4355 030044  022737  000010  002250            CMP     #10,FLAG                ;LOOK IF 10 REPORT
4356 030052  001314                            BNE     RXACK
4357
4358 030054                                    ESCAPE  TST
4359
4360
4361
4362
4363
4364 030060                            RXAEND:
4365 030060                            ENDTST
```

```
4367 030062                             BADHEAD
                                        ;********************** TEST10 **************
4368                                    ;TRANSMIT DIFFERENT FRAMES OF VARIOUS LENGTH IN EXTERNAL LOOP BACK
4369                                    ;MODE ON CHANNEL B
4370 030062                             BADHEAD
                                        ;********************** TEST10 **************
4371
4372
4373
4374
4375
4376
4377
4378 030062                             STARS 1
4379                                    ;
4380                                    ;AT BEGINNING OF TEST ,CHECK IF LOOP BACK CONNECTORS ARE INSTALLED
4381                                    ;OR NOT:IF NOT INSTALLED = EXIT TEST AND GIVE ERROR MESSAGE
4382                                    ;                          ***********************************
4383                                    ;
4384                                    ;
4385                                    ;
4386                                    ;
4387                                    ;
4388                                    ;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
4389                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED (56KB)
4390                                    ;
4391                                    ;
4392                                    ;
4393                                    ;DCT11 EXECUTE THE TRANSFER IN EXTERNAL MODE ON CHA  AND WRITE BACK
4394                                    ;IN RX TABLE
4395                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
4396                                    ;RX TABLE =TX TABLE
4397                                    ;
4398                                    ;
4399                                    ;PARAMETERS SELECTION:
4400                                    ;       SEL2=   TX TABLE ADDRESS
4401                                    ;       SEL4=   TX TABLE LENGTH
4402                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
4403                                    ;       BSEL7=      "           "   RX   "
4404                                    ;       SEL12=  RX TABLE ADDRESS
4405                                    ;       SEL14=  SPEED SELECTION
4406                                    ;       SEL16=  ERROR STATUS
4407                                    ;       BSEL0=  TEST STATUS
4408                                    ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
4409                                    ;                                       >0 IF TX>RX,<0 IF TX<RX
4410                                    ;
4411                                    ;
4412                                    ;
4413                                    ;
4414                                    ;TEST STATUS DESCRIPTION:
4415                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
4416                                    ;       BSEL0=  200     =TIMEOUT ERROR
4417                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
4418                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
4419                                    ;
4420                                    ;
4421                                    ;
```

```
4422                    ;ERROR STATUS DESCRIPTION:
4423                    ;
4424                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
4425                    ;
4426                    ;
4427                    ;       SEL16=  BIT14=1 =FCS ERROR
4428                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
4429                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERRUPT ERROR
4430                    ;       SEL16=  BIT7 =1 =RX ABORT ERROR
4431                    ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
4432                    ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
4433                    ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
4434                    ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
4435                    ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USED
4436                    ;                                               ONLY DURING SELF TEST)
4437                    ;
4438                    ;
4439                    ;
4440                    ;MICRO DIAG TEST DESCRIPTION:
4441                    ;TEST 43        =TRANSMIT VARIOUS LENGTHFRAME AT 56 KBAUDS SPEED ON CHANNEL B
4442                    ;                       IN EXTERNAL LOOP BACK MODE
4443                    ;
4444                    ;
4445                    ;
4446                    ;
4447                    ;CAUTION:
4448                    ;--------
4449                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4450                    ;
4451                    ;
4452                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4453                    ;
4454                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4455                    ;EXTERNAL LOOP BACK CONECTOR
4456                    ;
4457                    ;
4458                    ;
4459                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4460                    ;MODEM CABLE.
4461                    ;
4462                    ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
4463                    ;MODEM CABLE.
4464                    ;CAUTION:
4465                    ;--------
4466                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4467                    ;
4468                    ;
4469                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4470                    ;
4471                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4472                    ;EXTERNAL LOOP BACK CONECTOR
4473                    ;
4474                    ;
4475                    ;
4476                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4477                    ;MODEM CABLE.
4478                    ;
```

```
                                        ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
                                        ;MODEM CABLE.
                                        ;
                                        ;
                                        ;
                                        ;
4485 030062                             STARS 1
```

```
4487 030062                          BGNTST
4488 030062  004737  014400                  JSR     PC,CLRKMV
4489 030066  004737  014600                  JSR     PC,CKKMV        ;LOOK IF LOOP BACK CON INSTALLED?
4490
4491
4492 030072  005737  012474                  TST     LOOP            ;IS LOOP BIT=1?
4493 030076  001412                          BEQ     BGNTXD          ;YES GO ON TEST
4494 030100                                  PRINTF  #MLOOP          ;NO LOOP BACK PLUGGED IN.THE MODULE
4495                                                                 ;WILL NOT BE TESTED IS EXTERNAL LOOP
4496
4497 030120  000137  030552                  JMP     RXDEND
4498
4499
4500
4501 030124  004737  014502          BGNTXD: JSR     PC,MAINM1       ;SET MAINT MODE
4502
4503 030130  012703  000004                  MOV     #4,R3           ;SELECT RANDOM PATTERN
4504 030134  012737  000174  012414          MOV     #KB56,TSPEED    ;SELECT SPEED
4505
4506 030142  012737  000001  012416  TXDTAR: MOV     #1,LENGTH       ;1ST TABLE LENGTH
4507
4508 030150  013704  012416          TXDBGN: MOV     LENGTH,R4
4509 030154  012702  002370                  MOV     #TTABLE,R2
4510 030160                                  BREAK
4511 030162  004737  013160          10$:    JSR     PC,GENER        ;WRITE TABLE
4512 030166  013722  012400                  MOV     DATA,(R2)+
4513 030172  005304                          DEC     R4
4514 030174  001372                          BNE     10$
4515
4516
4517 030176  013704  012416                  MOV     LENGTH,R4       ;CLEAR RX TABLE
4518 030202  012702  006370                  MOV     #RTABLE,R2
4519 030206  005022                  20$:    CLR     (R2)+
4520 030210  005304                          DEC     R4
4521 030212  001375                          BNE     20$
4522
4523
4524
4525
4526
4527 030214  013777  012414  162246          MOV     TSPEED,@KMVP14  ;SEND TX SPEED
4528 030222  012777  002370  162226          MOV     #TTABLE,@KMVP02 ;  "  TX TABLE ADDRESS
4529 030230  013777  012416  162222          MOV     LENGTH,@KMVP04  ;  "  "     "  LENGTH
4530 030236  012777  006370  162222          MOV     #RTABLE,@KMVP12 ;SEND RX TABLE ADDRESS
4531 030244  005077  162212                  CLR     @KMVP06         ;CLR EXTENDED ADDRESS
4532
4533
4534
4535
4536
4537 030250  004537  014552          1$:     JSR     R5,TSTNUB
4538 030254  000043                          .WORD   43              ;DO TEST 43= CHB TEST
4539
4540
4541
4542 030256                          2$:     WAITB   0,3             ;WAIT FOR TEST EXECUTION
4543
```

```
4544
4545 030276  004737  013102                    JSR     PC,TSTERR               ;CHECK BSEL0
4546
4547 030302  000427                            BR      6$                      ;TEST OK CHECK RX TABLE
4548 030304  000402                            BR      3$                      ;TIMEOUT ERROR
4549 030306  000401                            BR      3$                      ;NO KMV11 ANSWER
4550 030310  000410                            BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
4551
4552
4553 030312                          3$:       ERRHRD  36,EM0004               ;NO KMV11 ANSWER
4554 030322  004737  012740                    JSR     PC,CHKMAX               ;TOO MANY ERROR, DROP IF YES
4555 030326                                    ESCAPE  TST
4556
4557
4558 030332                          4$:                                       ;ERROR DURING TEST READ ERROR STATUS
4559                                                                           ;TO CHECK WHICH ONE
4560
4561 030332  017737  162134  012424            MOV     @KMVP16,STAERR          ;READ ERROR STATUS
4562
4563 030340  017737  162120  012426            MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
4564
4565 030346                                    ERRHRD  37,EM0020,PRSTER        ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
4566                                                                           ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
4567 030356                                    ESCAPE  TST
4568
4569
4570
4571
4572
4573 030362  012702  002370          6$:       MOV     #TTABLE,R2
4574 030366  012705  006370                    MOV     #RTABLE,R5
4575 030372  013704  012416                    MOV     LENGTH,R4
4576 030376  022225                  RXDCK:    CMP     (R2)+,(R5)+
4577 030400  001015                            BNE     RXDERR
4578 030402  005304                            DEC     R4
4579 030404  001374                            BNE     RXDCK
4580
4581 030406  062737  000400  012416            ADD     #400,LENGTH
4582 030414  022737  002000  012416            CMP     #2000,LENGTH
4583 030422  100252                            BPL     TXDBGN
4584
4585 030424  005303                            DEC     R3
4586 030426  001245                            BNE     TXDTAR
4587 030430  000137  030552                    JMP     RXDEND
4588
4589
4590
4591 030434  162705  000002          RXDERR:   SUB     #2,R5
4592 030440  162702  000002                    SUB     #2,R2
4593
4594 030444  011237  012410                    MOV     (R2),TXDATA
4595 030450  011537  012412                    MOV     (R5),RXDATA
4596
4597 030454  005737  002250                    TST     FLAG                    ;LOOK IF 1ST ERROR
4598 030460  001014                            BNE     30$
4599
4600 030462                                    ERRHRD  38,EM0015,PFRAME        ;DATA CMP ERROR
```

```
4601 030472  005237  002250                    INC     FLAG
4602 030476  062702  000002                    ADD     #2,R2                   ;POINT NEXT ADDRESS
4603 030502  062705  000002                    ADD     #2,R5
4604 030506  000137  030376                    JMP     RXDCK
4605
4606 030512                            30$:    ERRHRD  38,0,PRAMEF             ;SHORT REPORT
4607 030522  005237  002250                    INC     FLAG
4608 030526  062702  000002                    ADD     #2,R2
4609 030532  062705  000002                    ADD     #2,R5                   ;POINT NEXT ADDRESS
4610 030536  022737  000010  002250            CMP     #10,FLAG                ;LOOK IF 10 REPORT
4611 030544  001314                            BNE     RXDCK
4612
4613 030546                                    ESCAPE  TST
4614
4615
4616
4617
4618
4619 030552                            RXDEND:
4620 030552                            ENDTST
```

```
 030554                    BADHEAD
                           ;************************ TEST11 ***************
                           ;TEST MODEM SIGNALS CCITT 108 AND CCITT 107 IN EXTERNAL LOOP BACK MODE
 030554                    BADHEAD
                           ;************************ TEST11 ***************






 030554                    STARS 1
                           ;CCITT 108/2 A IS BIT 7 IN 8255 CHIP ,PORT B = ADDRESS 130012
                           ;    "        B      " 6      "    "     " B      "     130012
                           ;    "   107  A      " 5      "    "     " A      "     130000
                           ;    "   107  B      " 3      "    "     " A      "     130000
                           ;
                           ;
                           ;QBUS WRITE CCITT 108A AND B ,AND READ BACK CCITT 107A/B
                           ;
                           ;
                           ;CAUTION:
                           ;--------
                           ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
                           ;
                           ;
                           ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BE INSTALLED
                           ;
                           ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
                           ;EXTERNAL LOOP BACK CONECTOR
                           ;
                           ;
                           ;
                           ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
                           ;MODEM CABLE.
                           ;
                           ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
                           ;MODEM CABLE.
                           ;
                           ;
                           ;
                           ;
 030554                    STARS 1

```

```
4667 030554                                BGNTST
4668 030554  004737  014400                       JSR     PC,CLRKMV          ;CLEAR KMV11 REGISTERS
4669 030560  004737  014600                       JSR     PC,CKKMV           ;LOOK IF KMV11A OR B AND IF LOOP BACK
4670
4671 030564  005737  012474                       TST     LOOP               ;LOOK IF LOOP BACK?
4672 030570  001412                             BEQ     MOD108             ;YES GO ON
4673 030572                                      PRINTF  #MLOOP             ;NO LOOP BACK PLUGGED .THE MODULE
4674                                                                         ;WILL NOT BE TESTED IN EXTERNAL LOOP
4675
4676 030612  000137  031012                      JMP     MODEND
4677
4678
4679
4680 030616  004737  014502          MOD108:     JSR     PC,MAINM1
4681
4682
4683 030622  012737  000100  030636              MOV     #100,MODWR1+6      ;WRITE TTL 1088
4684
4685
4686
4687 030630  004537  014702          MODWR1:     JSR     R5,WRITE           ;WRITE KMV REG ADDRESS 130012
4688 030634  130012                             .WORD   130012
4689 030636  000000                             .WORD   0                  ;DATA TO WRITE
4690
4691 030640                                      WAITA   0
4692
4693
4694 030652  004537  014730                      JSR     R5,READ            ;READ KMV ADDRESS 130000
4695 030656  130000                             .WORD   130000
4696
4697
4698 030660  042737  177767  012374              BIC     #177767,BAD        ;MASK UNUSED BITS
4699 030666  022737  000010  012374              CMP     #10,BAD            ;CMP 108B AND 107B
4700 030674  001036                             BNE     MODERB             ;REPORT ERROR IF BAD
4701
4702
4703
4704 030676  012737  000200  030712              MOV     #200,MODWR2+6      ;WRITE TTL 108A
4705
4706
4707
4708 030704  004537  014702          MODWR2:     JSR     R5,WRITE           ;WRITE KMV REG ADDRESS 130012
4709 030710  130012                             .WORD   130012
4710 030712  000000                             .WORD   0                  ;DATA TO WRITE
4711
4712 030714                                      WAITA   0
4713
4714
4715 030726  004537  014730                      JSR     R5,READ            ;READ KMV ADDRESS 130000
4716 030732  130000                             .WORD   130000
4717
4718
4719 030734  042737  177737  012374              BIC     #177737,BAD        ;MASK BIT BUT CCITT 107A/B
4720 030742  022737  000040  012374              CMP     #40,BAD            ;CMP 108A AND 107A
4721 030750  001420                             BEQ     MODEND             ;OK EXIT TEST
4722
4723
```

```



030752                              MODERA: ERRHRD  40,EM0030              ;DATA CMP ERROR BETWEEN 107 AND 108
                                                                           ; ON CHANNEL A
030762  004737  012740                      JSR     PC,CHKMAX              ;DROP IF TOO MANY ERROR
030766                                      ESCAPE TST



030772                              MODERB: ERRHRD  41,EM0130              ;DATA CMP ERROR BETWEEN 107 AND 108
                                                                           ; ON CHANNEL B
031002  004737  012740                      JSR     PC,CHKMAX              ;DROP IF TOO MANY ERROR
031006                                      ESCAPE TST




031012                              MODEND:
031012                              ENDTST
```

```
     031014                  BADHEAD
                             ;*********************** TEST12 ***************
                             ;TEST MODEM SIGNALS ON KV11 B MODULE IN EXTERNAL LOOP BACK ON CHANEL A AND B
     031014                  BADHEAD
                             ;*********************** TEST12 ***************





     031014                  STARS 1
                             ;HOST SET TEST NUMBER 34 OR 35 (CHA OR CHB TEST)
                             ;DCT11 TEST MODEM SIGNAL 105,106,109 ON CHANEL A(TEST NB34) FIRST AND THEN
                             ;ON CHANEL B(TEST NB 35) BY SETTING AND CLEARING BIT 105 AND TESTING
                             ;BIT 106,AND 109
                             ;
                             ;IF TEST =OK,DCT11 CLEAR BSEL0
                             ;IF ERROR SET 100 IN BSEL0 AND REPORT ERROR
                             ;
                             ;
                             ;ERROR REPORT DESCRIPTION:
                             ;SEL2 INDICATE WHICH MODEM SIGNAL IS TESTED
                             ;SEL4 INDICATE THE RESULT OF THE TEST
                             ;SEL10 INDICATE IF IT WAS DURING A CLEAR OR A SET OPERATION
                             ;
                             ;SEL 10 BIT 1=0 INDICATE A CLEAR OPERATION ON TESTED MODEM SIGNAL
                             ;            =1     ''      SET      ''       ''    ''     ''      ''
                             ;
                             ;
                             ;
                             ;SEL2 AND SEL4 FORMAT:
                             ;       -----------------------------------------------------------------
                             ;       /       /       /       /       / 109   /  106  /       / 105   /
                             ;       -----------------------------------------------------------------
                             ;        BIT 7   BIT 6   BIT5    BIT4    BIT3    BIT2    BIT1    BIT0
                             ;
                             ;
                             ;MODEM SIGNAL LINK:
                             ;
                             ;
                             ;
                             ;       CCITT 105 (RTS)                           CCITT 106 (CTS)
                             ;       --------------------------------------------------------------
                             ;                                        /
                             ;                                        /
                             ;                                        /          CCITT 109 (CD)
                             ;                                        ------------------------------
                             ;
                             ;
                             ;
                             ;
                             ;
                             ;MICRO TEST DESCRIPTION:
                             ;TEST 34:MODEM SIGNAL ON CHANEL A
                             ;TEST 35:MODEM SIGNAL ON CHANEL B
                             ;
```

```
4800                                    ;
4801                                    ;CAUTION:
4802                                    ;--------
4803                                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4804                                    ;
4805                                    ;
4806                                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4807                                    ;
4808                                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4809                                    ;EXTERNAL LOOP BACK CONNECTOR
4810                                    ;
4811                                    ;
4812                                    ;
4813                                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4814                                    ;MODEM CABLE.
4815                                    ;
4816                                    ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
4817                                    ;MODEM CABLE.
4818                                    ;
4819 031014                            STARS 1
4820
4821
```

HARDWARE TESTS

```
4823 031014                                BGNTST
4824 031014  004737  014400                       JSR     PC,CLRKMV          ;CLEAR ALL REGISTERS
4825 031020  004737  014600                       JSR     PC,CKKMV           ;TEST IF LOOP BACK CONNECTOR
4826
4827 031024  005737  012474                       TST     LOOP
4828 031030  001412                              BEQ     1$                 ;LOOP BACK PRESENT GO ON
4829
4830 031032                                      PRINTF  #MLOOP             ;NO LOOP BACK PLUGGED IN .THE MODULE
4831                                                                         ;WILL NOT BE TESTED IS EXTERNAL LOOP
4832
4833 031052  000137  031012                      JMP     MODEND             ;GO TO FOLLOWING TEST
4834
4835
4836 031056                                1$:
4837 031056                                BGNSUB
4838
4839 031060  004737  014502                       JSR     PC,MAINM1          ;SET MAINTENANCE MODE
4840 031064  004537  014552                       JSR     R5,TSTNUB
4841 031070  000034                              .WORD   34                 ;SEND TEST 34(MODEM SIGNAL ON CHA)
4842
4843 031072                                      WAITB   0,2
4844
4845 031112  004737  013102                       JSR     PC,TSTERR          ;CHECK TEST RESULT
4846 031116  000432                              BR      3$                 ;TEST OK GO ON
4847 031120  000402                              BR      4$                 ;TIMEOUT
4848 031122  000401                              BR      4$                 ;NO TEST ANSWER
4849 031124  000410                              BR      5$                 ;ERROR DURING TEST ,LOOK WHICH ONE
4850
4851
4852
4853 031126                                4$:   ERRHRD  42,EM0004          ;NO ANSWER
4854 031136  004737  012740                       JSR     PC,CHKMAX          ;DROP IF TOO MANY ERROR
4855 031142                                      ESCAPE SUB
4856
4857 031146  017737  161304  002272        5$:   MOV     @KMVP02,GOOD       ;READ WHICH SIGNAL WAS TESTED
4858 031154  017737  161300  012374              MOV     @KMVP04,BAD        ;   "     "     IS THE RESULT OF TEST
4859 031162  017737  161276  012400              MOV     @KMVP10,DATA       ;READ SIGAL VALUE
4860
4861 031170                                      ERRHRD  43,EM0032,PMODEM   ;REPORT ERROR
4862 031200                                      ESCAPE  SUB
4863
4864 031204                                3$:
4865 031204                                ENDSUB
4866
4867
4868 031206                                BGNSUB
4869
4870 031210  004737  014502                       JSR     PC,MAINM1          ;SET MAINTENANCE MODE
4871 031214  004537  014552                       JSR     R5,TSTNUB
4872 031220  000035                              .WORD   35                 ;SEND TEST 35(MODEM SIGNAL ON CHB)
4873
4874 031222                                      WAITB   0,2
4875
4876 031242  004737  013102                       JSR     PC,TSTERR          ;CHECK TEST RESULT
4877 031246  000432                              BR      3$                 ;TEST OK GO ON
4878 031250  000402                              BR      4$                 ;TIMEOUT
4879 031252  000401                              BR      4$                 ;NO TEST ANSWER
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 79-1
HARDWARE TESTS

   4880 031254  000410                          BR      5$                      ;ERROR DURING TEST ,LOOK WHICH ONE
   4881
   4882
   4883
   4884 031256                          4$:     ERRHRD  44,EM0004               ;NO ANSWER
   4885 031266  004737  012740                  JSR     PC,CHKMAX               ;DROP IF TOO MANY ERROR
   4886 031272                                  ESCAPE SUB
   4887
   4888 031276  017737  161154  002272  5$:     MOV     @KMVP02,GOOD            ;READ WHICH SIGNAL WAS TESTED
   4889 031304  017737  161150  012374          MOV     @KMVP04,BAD             ;  ''     ''    IS THE RESULT OF TEST
   4890 031312  017737  161146  012400          MOV     @KMVP10,DATA            ;READ SIGAL VALUE
   4891
   4892 031320                                  ERRHRD  45,EM0034,PMODEM        ;REPORT ERROR
   4893 031330                                  ESCAPE  SUB
   4894
   4895 031334                          3$:
   4896 031334                          ENDSUB
   4897
   4898
   4899 031336                          ENDTST
```

```
 4901
 4902
 4903
 4904
 4905
 4906
 4907                                .SBTTL HARDWARE PARAMETER CODING SECTION
 4908
 4909
 4910
 4911                                ;////////////////////////////////////////////////////////////////////////
 4912                                ;/ THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
 4913                                ;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
 4914                                ;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
 4915                                ;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
 4916                                ;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
 4917                                ;/ WITH THE OPERATOR.
 4918                                ;////////////////////////////////////////////////////////////////////////
 4919
 4920 031340                                BGNHRD
 4921
 4922 031342                                GPRMA   ADDRES,0,0,60000,177776,YES
 4923 031352                                GPRMA   VECTOR,2,0,0,674,YES
 4924 031362                                GPRMD   PRIRTY,4,0,7000,4,7,YES
 4925 031374                                ENDHRD
 4926
 4933
 4934
 4935 031374    115     111     103  ADDRES: .ASCIZ  /MICRO-CPU  CSR ADDRESS : /
      031377    122     117     055
      031402    103     120     125
      031405    040     040     103
      031410    123     122     040
      031413    101     104     104
      031416    122     105     123
      031421    123     040     072
      031424    040     000
 4936 031426    115     111     103  VECTOR: .ASCIZ  /MICRO-CPU VECTOR ADDRESS : /
      031431    122     117     055
      031434    103     120     125
      031437    040     126     105
      031442    103     124     117
      031445    122     040     101
      031450    104     104     122
      031453    105     123     123
      031456    040     072     040
      031461    000
 4937 031462    115     111     103  PRIRTY: .ASCIZ  /MICRO-CPU PRIORITY LEVEL : /
      031465    122     117     055
      031470    103     120     125
      031473    040     120     122
      031476    111     117     122
      031501    111     124     131
      031504    040     114     105
      031507    126     105     114
      031512    040     072     040
      031515    000
```

HARDWARE PARAMETER CODING SECTION

```
4938
4939                                    .EVEN
4940
4941
4942
4943
4944
4945
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION


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

031516                                  BGNSFT



031520                                  ENDSFT
```

```
 4983          032000                          .=<.!376>+2             ; END OF PAGE
 4984
 4985
 4986 032000                            $PATCH::
 4987 032000                                    .BLKW   60
 4988
 4995
 4996 032140                                    LASTAD
      032144                            L$LAST::
 4997 032144                                    ENDMOD
 4998
 4999
```

```
5001
5002
5015
5016 032144                                    BGNSETUP        1
5017 032144                                    BGNPTAB
5018 032150  177000                            .WORD   177000
5019 032152  000300                            .WORD   300
5020 032154  004000                            .WORD   4000
5021 032156  000001                            .WORD   1
5022 032160                                    ENDPTAB
5023 032160                                    ENDSETUP
5024
5025
5026
5027
5028
5029         000001                  .END
```

SYMBOL TABLE

```
ABORT   023524          CKSEL0  013620          DATA1 = 052525 G        F$END = 000041          G$RADA= 000140
ADDR    002362          CLRKMV  014400          DATA2 = 125252 G        F$HARD= 000004          G$RADB= 000000
ADDRES  031374          COUNT   002356          DELCT1  002266          F$HW  = 000013          G$RADD= 000040
ADR   = 000020 G        C$AU  = 000052          DELCT2  002270          F$INIT= 000006          G$RADL= 000120
ASSEMB= 000010          C$AUTO= 000061          DFPTBL  002164 G        F$JMP = 000050          G$RADO= 000020
BAD     012374          C$BRK = 000022          DIAGMC= 000000          F$MOD = 000000          G$XFER= 000004
BDDAT   002366          C$BSEG= 000004          DROPD   023740          F$MSG = 000011          G$YES = 000010
BDRGEN  024516          C$BSUB= 000002          EF.CON= 000036 G        F$PROT= 000021          HELP  = 000000
BDROK0  024742          C$CEFG= 000045          EF.NEW= 000035 G        F$PWR = 000017          HOE   = 100000 G
BDROK1  025210          C$CLCK= 000062          EF.PWR= 000034 G        F$RPT = 000012          IBE   = 010000 G
BGNTXA  027426          C$CLEA= 000012          EF.RES= 000037 G        F$SEG = 000003          IDU   = 000040 G
BGNTXD  030124          C$CLOS= 000035          EF.STA= 000040 G        F$SOFT= 000005          IER   = 020000 G
BIT0  = 000001 G        C$CLP1= 000006          EM0001  015120          F$SRV = 000010          INIFLG  012432
BIT00 = 000001 G        C$CVEC= 000036          EM0002  015215          F$SUB = 000002          INTFLG  012372
BIT01 = 000002 G        C$DCLN= 000044          EM0003  015261          F$SW  = 000014          INTTX   025244
BIT02 = 000004 G        C$DODU= 000051          EM0004  015347          F$TEST= 000001          ISR   = 000100 G
BIT03 = 000010 G        C$DRPT= 000024          EM0006  015376          GDDAT   002364          IXE   = 004000 G
BIT04 = 000020 G        C$DU  = 000053          EM0007  015440          GDREV   012430          I$AU  = 000041
BIT05 = 000040 G        C$EDIT= 000003          EM0010  015531          GENER   013160          I$AUTO= 000041
BIT06 = 000100 G        C$ERDF= 000055          EM0011  015604          GENER1  013302          I$CLN = 000041
BIT07 = 000200 G        C$ERHR= 000056          EM0012  015664          GENEX   013440          I$DU  = 000041
BIT08 = 000400 G        C$ERRO= 000060          EM0013  015762          GENINC  013432          I$HRD = 000041
BIT09 = 001000 G        C$ERSF= 000054          EM0014  016262          GENISH  013310          I$INIT= 000041
BIT1  = 000002 G        C$ERSO= 000057          EM0015  016333          GENO    025170          I$MOD = 000041
BIT10 = 002000 G        C$ESCA= 000010          EM0016  016432          GENOUT  024722          I$MSG = 000041
BIT11 = 004000 G        C$ESEG= 000005          EM0017  016723          GENRAN  013312          I$PROT= 000040
BIT12 = 010000 G        C$ESUB= 000003          EM0020  017020          GENROT  013266          I$PTAB= 000041
BIT13 = 020000 G        C$ETST= 000001          EM0023  017115          GENR0   013254          I$PWR = 000041
BIT14 = 040000 G        C$EXIT= 000032          EM0024  017200          GENR1   013244          I$RPT = 000041
BIT15 = 100000 G        C$GETB= 000026          EM0027  017227          GENSEL  013176          I$SEG = 000041
BIT2  = 000004 G        C$GETW= 000027          EM0030  017301          GEN0    013216          I$SETU= 000041
BIT3  = 000010 G        C$GMAN= 000043          EM0032  017473          GEN1    013222          I$SFT = 000041
BIT4  = 000020 G        C$GPHR= 000042          EM0033  016204          GEN25   013236          I$SRV = 000041
BIT5  = 000040 G        C$GPLO= 000030          EM0034  017564          GEN52   013230          I$SUB = 000041
BIT6  = 000100 G        C$GPRI= 000040          EM0035  017655          GETPRM  023322          I$TST = 000041
BIT7  = 000200 G        C$INIT= 000011          EM0112  016034          GOOD    002272          J$JMP = 000167
BIT8  = 000400 G        C$INLP= 000020          EM0113  016132          GOOD0   002274          KB1.2 = 013224 G
BIT9  = 001000 G        C$MANI= 000050          EM0115  016527          GOOD1   002276          KB56  = 000174 G
BOE   = 000400 G        C$MEM = 000031          EM0116  016626          GOOD10  002306          KB64  = 000154 G
BRXCK   026122          C$MSG = 000023          EM0130  017376          GOOD12  002310          KB68  = 000146 G
BRXEND  026262          C$OPEN= 000034          END     023532          GOOD14  002312          KB70  = 000143 G
BRXERR  026144          C$PNTB= 000014          ERRBLK  002230 G        GOOD16  002314          KB72  = 000141 G
BSEL0   012376          C$PNTF= 000017          ERRCNT  002236          GOOD2   002300          KIND    012404
BSEL1   002340          C$PNTS= 000016          ERRMSG  002226 G        GOOD4   002302          KMTLVL  012452
BTXSTA  025674          C$PNTX= 000015          ERRNBR  002224 G        GOOD6   002304          KMVCSR  012454
BXCK  = ****** GX       C$QIO = 000377          ERRTYP  002222 G        G$CNTO= 000200          KMVLVL  012442
BXLBGN  026760          C$RUBU= 000007          EVL   = 000004 G        G$DELM= 000372          KMVP02  012456
BXLCK   027206          C$REFG= 000047          EXADDR  012370          G$DISP= 000003          KMVP04  012460
BXLEND  027362          C$RESE= 000033          E$END = 002100          G$EXCP= 000400          KMVP06  012462
BXLERR  027244          C$REVI= 000003          E$LOAD= 000035          G$HILI= 000002          KMVP10  012464
BXLTAR  026752          C$RFLA= 000021          FLAG    002250          G$LOLI= 000001          KMVP12  012466
CBSEL0  013652          C$RPT = 00C025          FTIME   002252          G$NO  = 000000          KMVP14  012470
CHANEL  012406          C$SEFG= 000046          F$AU  = 000015          G$OFFS= 000400          KMVP16  012472
CHKMAX  012740          C$SPRI= 000041          F$AUTO= 000020          G$OFSI= 000376          KMVV00  012440
CKALL   013710          C$SVEC= 000037          F$BGN = 000040          G$PRMA= 000001          KMVV02  012446
CKKMV   014600          C$TPRI= 000013          F$CLEA= 000007          G$PRMD= 000002          KMVV04  012444
CKREG   014156          DATA    012400          F$DU  = 000016          G$PRML= 000000          KMVV06  012450
```

SYMBOL TABLE

```
KMV11B  002000 G        L$STA   002030 G        MLOOP   017736          PRSEL0  021520 G        STAERR  012424
LENGTH  012416          L$SW    002260          MODEM1  021301          PRSTER  023006 G        SVCGBL= 000000
LOCK    002232          L$TEST  002114 G        MODEM2  021345          PRT11V  022526 G        SVCINS= 177777
LOE   = 040000 G        L$TIML  002014 G        MODEM3  021404          PSTACK  002242          SVCSUB= 177777
LOGDEV  002240          L$UIT   002262          MODEND  031012          PVECT   022470 G        SVCTAG= 177777
LOKFLG  012434          L$UNIT  002012 G        MODERA  030752          QV.FLG  012435          SVCTST= 177777
LOOP    012474          L10001  002174          MODERB  030772          RANCLC  013412          S$LSYM= 010000
LOT   = 000010 G        L10002  021554          MODWR1  030630          RANDN   002350          TFM36   015032
L$ACP   002110 G        L10003  021612          MODWR2  030704          RANEX   013430          TIM     015014
L$APT   002036 G        L10004  022120          MOD108  030616          RANGEN  013332          TSPEED  012414
L$AU    023772 G        L10005  022376          MRAMEF  021447          RANMTA  002346          TSTERR  013102
L$AUT   002070 G        L10006  022434          MREG0   020062          RANSEC  013416          TSTNUB  014552
L$AUTO  023614 G        L10007  022466          MREG10  020302          RANSEL  002344          TTABLE  002370
L$CCP   002106 G        L10010  022524          MREG12  020346          RANST   002342          TXABGN  027456
L$CLEA  023706 G        L10011  022562          MREG14  020412          RAN1    013344          TXATAR  027450
L$CO    002032 G        L10012  022650          MREG16  020456          RAN2    013362          TXDATA  012410
L$DEPO  002011 G        L10013  022752          MREG2   020126          RAN4    013420          TXDBGN  030150
L$DESC  002174 G        L10014  023004          MREG4   020172          READ    014730          TXDTAR  030142
L$DESP  002076 G        L10015  023062          MREG6   020236          REGADR  012476          TXLBGN  026320
L$DEVP  002060 G        L10016  023120          MSEL0   020014          REVPRO  024126          TXLTAR  026312
L$DISP  002132 G        L10017  023126          MSTER1  021204          ROMMAP  023774          TXSTAR  025250
L$DLY   002116 G        L10020  023532          MSTER2  021236          RSTREG  013540          T$ARGC= 000001
L$DTP   002040 G        L10021  023704          MT11V   020761          RTABLE  006370          T$CODE= 002032
L$DTYP  002034 G        L10022  023710          MVECT   020705          RTCLK   024176          T$ERRN= 000055
L$DU    023712 G        L10023  023770          NERRS   013032          RUNNIN  023534          T$EXCP= 000000
L$DUT   002072 G        L10024  023772          NEXT    023242          RXACK   027704          T$FLAG= 000040
L$DVTY  012676 G        L10025  024114          NUB     012420          RXAEND  030060          T$FREE= 032160
L$EF    002052 G        L10026  024164          NUMBER  002360          RXAERR  027742          T$GMAN= 000000
L$ENVI  002044 G        L10027  024474          O$APTS= 000000          RXCK    025476          T$HILI= 000007
L$ERRT  002222 G        L10030  025212          O$AU  = 000000          RXCNT   012422          T$LAST= 000001
L$ETP   002102 G        L10031  024742          O$BGNR= 000000          RXDATA  012412          T$LOLI= 000004
L$EXP1  002046 G        L10032  025210          O$BGNS= 000000          RXDCK   030376          T$LSYM= 010000
L$EXP4  002064 G        L10033  025636          O$DU  = 000001          RXDEND  030552          T$LTNO= 000014
L$EXP5  002066 G        L10034  026262          O$ERRT= 000000          RXDERR  030434          T$NEST= 177777
L$HARD  031342 G        L10035  026722          O$GNSW= 000001          RXEND   025636          T$NS0 = 000000
L$HIME  002120 G        L10036  027362          O$POIN= 000001          RXERR   025520          T$NS1 = 000005
L$HPCP  002016 G        L10037  030060          O$SETU= 000001          RXLCK   026546          T$NS2 = 000002
L$HPTP  002022 G        L10040  030552          PADFLT  023064 G        RXLEND  026722          T$PCNT= 000000
L$HW    002164 G        L10041  031012          PBSEL0  022400 G        RXLERR  026604          T$PTAB= 010050
L$ICP   002104 G        L10042  031336          PFRAME  022564 G        SAVE4   002254          T$PTHV= 000001
L$INIT  023130 G        L10043  031204          PINTR   022436 G        SAVE6   002256          T$PTNU= 000001
L$LADP  002026 G        L10044  031334          PMODEM  022652 G        SAVPC   002246          T$SAVL= 177777
L$LAST  032144 G        L10045  031374          PNT   = 001000 G        SAVPC1  002352          T$SEGL= 177777
L$LOAD  002100 G        L10046  031520          PRALL   021614 G        SAVREG  013460          T$SIZE= 000006
L$LUN   002074 G        L10047  032150          PRAMEF  022754 G        SAVSP   002244          T$SUBN= 000002
L$MREV  002050 G        L10051  032160          PRI   = 002000 G        SAVSTA  002354          T$TAGL= 177777
L$NAME  002000 G        MAINM1  014502          PRINT   021556 G        SEL0    002316          T$TAGN= 010052
L$PRIO  002042 G        MAINT0= 054000 G        PRIRTY  031462          SEL1    002320          T$TEMP= 000000
L$PROT  002122 G        MAINT1= 044000 G        PRI00 = 000000 G        SEL10   002330          T$TEST= 000014
L$PRT   002112 G        MAXERR  002234          PRI01 = 000040 G        SEL12   002332          T$TSTM= 177777
L$REPP  002062 G        MAXPRI= 000340 G        PRI02 = 000100 G        SEL14   002334          T$TSTS= 000001
L$REV   002010 G        MBSEL0  020556          PRI03 = 000140 G        SEL16   002336          T$$AU = 010024
L$RPT   023122 G        MCLR  = 040000 G        PRI04 = 000200 G        SEL2    002322          T$$AUT= 010021
L$SOFT  031520 G        MFRAM1  021043          PRI05 = 000240 G        SEL4    002324          T$$CLE= 010022
L$SPC   002056 G        MFRAM2  021121          PRI06 = 000300 G        SEL6    002326          T$$DAT= 010051
L$SPCP  002020 G        MINT    020522          PRI07 = 000340 G        SETUP   023234          T$$DU = 010023
L$SPTP  002024 G        MINTR   020620          PRREG   022122 G        SSTACK  012676          T$$HAR= 010045
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  26-APR-83 14:51  PAGE 83-3
SYMBOL TABLE

T$$HW = 010001          T$$TES= 010042          T4      024476 G        UAM   = 000200 G        WRITE   014702
T$$INI= 010020          T1      023774 G        T4.1    024476          UNIT    002264          X$ALWA= 000000
T$$MSG= 010016          T10     030062 G        T4.2    024744          UUT     012436          X$FALS= 000040
T$$PC = 000001          T11     030554 G        T5      025214 G        VECT    012402          X$OFFS= 000400
T$$PRO= 010000          T12     031014 G        T6      025640 G        VECTOR  031426          X$TRUE= 000020
T$$PTA= 010050          T12.1   031056          T7      026264 G        WAIT1   012726          $LSTIN= 000000
T$$RPT= 010017          T12.2   031206          T8      026724 G        WAIT2   012706          $LSTTA= 000000
T$$SOF= 010046          T2      024116 G        T9      027364 G        WRDCNT  012426          $PATCH  032000 G
T$$SUB= 010044          T3      024166 G

. ABS.  032160     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29104 WORDS  ( 114 PAGES)
DYNAMIC MEMORY:  21924 WORDS  ( 84 PAGES)
ELAPSED TIME:  00:04:21
VKMEB0.BIN,VKMEB0=[64,3]LIBA.MLB/ML,[64,6]VKMEB0
```