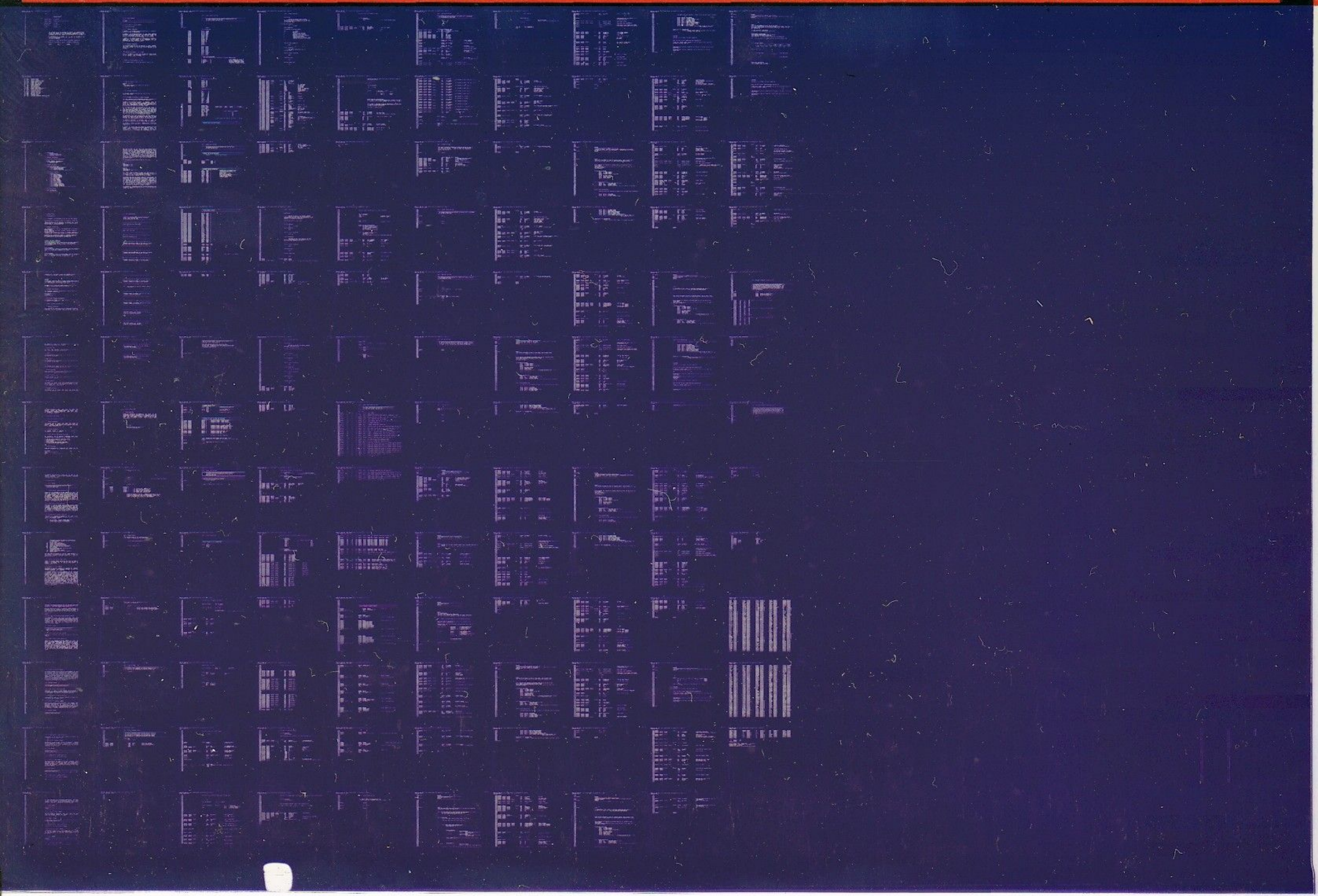
KMV11-B

LINE CNT DIAG
CVKMEA0

AH-T378A-MC
FICHE 1 OF 1



```
.REM @
```

IDENTIFICATION

PRODUCT CODE: AC-T377A-MC

PRODUCT NAME: CVKMEAO KMV11B LINE CNT DIAG

PRODUCT DATE: JAN 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 UNIBUS 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 ,DROP EXTERNAL TEST AND REPORT ERROR)

                                DIAGNOSTIC WILL AUTOMATICALY DETECT IF LOOPBACK CONNECTOTS
                                ARE PLUGGED OR NOT.(IF NO LOOPBACK,EXTERNAL LOOP TEST WILL
                                BE DROPPED AND AN ERROR MESAGE WILL BE REPORTED).



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


                                RS422 LOOP BACK:
                                TO TEST COMPLETELY A KMV11 B IN RS422 MODE ,RUN THIS DIAGNOSTIC
                                WHITH 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
                                WHITH  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 CORRESPONDING TO 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 KMV11A/B
                                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 220 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
VKMEA0
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.  SEF  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>
****

```
532
533
534
535
536
537                         6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
538
539                         <FLAG-LIST>  IS  AS  IN  THE  START COMMAND, BUT UNSPECIFIED
540                         FLAGS RETAIN THEIR CURRENT VALUE.
541
542
543                         6.3.4.2 EFFECT OF PROCEED COMMAND
544
545                         PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE.   COMMAND
546                         MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR.  THE EFFECT
547                         OF THE  COMMAND  IS  TO  BEGIN  EXECUTION  AT  THE  LOCATION
548                         FOLLOWING  THE  ERROR  CALL.   NEITHER HARDWARE NOR SOFTWARE
549                         PARAMETERS MAY BE ALTERED.
550
551
552                         6.3.5 ADD COMMAND
553
554                         ************************************************************
555                         ADD/UNITS:<UNIT-LIST>
556                         ************************************************************
557
558
559                         6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>
560
561                         <UNIT-LIST> IS AS IN THE RESTART COMMAND.
562
563
564                         6.3.5.2 EFFECT OF ADD COMMAND
565
566                         THE UNITS SPECIFIED ARE ADDED TO THE  TEST  SEQUENCE.   EACH
567                         UNIT  MUST  HAVE  A  P-TABLE  IN  MEMORY  DUE  TO AN EARLIER
568                         HARDWARE DIALOGUE.  THIS  COMMAND  MUST  BE  FOLLOWED  BY  A
569                         RESTART  OR  CONTINUE.   THE UNITS SWITCH MUST BE SPECIFIED.
570                         THE ADD COMMAND IS  MEANINGFUL  ONLY  FOR  UNITS  THAT  WERE
571                         PREVIOUSLY DROPPED.
572
573
574                         6.3.6 DROP COMMAND
575
576                         ************************************************************
577                         DRO(P)/UNITS:<UNIT-LIST>
578                         ************************************************************
579
580
581                         6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)
582
583                         <UNIT-LIST> IS AS IN THE RESTART COMMAND.
584
585
586                         6.3.6.2 EFFECT OF DROP COMMAND
```

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

6.3.7 PRINT COMMAND

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

6.3.7.1 EFFECT OF PRINT COMMAND

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

6.3.8 DISPLAY COMMAND

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

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

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

6.3.8.2 EFFECT OF DISPLAY COMMAND

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

6.3.9 FLAGS COMMAND

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

6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

6.3.10 ZFLAGS COMMAND

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

6.3.10.1 EFFECT OF ZFLAGS COMMAND

ALL FLAGS ARE CLEARED.

6.3.11 CONTROL CHARACTERS

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

A CONTROL Z (Z) ENTERED DURING ONE OF THE THREE OPERATOR DIALOGUES- 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 FRIORITY 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 *********************
                        *
                        *REAL TIME CLOCK TEST
                        *
                        ****************************************************


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


                        ********************* TEST 5 *********************
                        *
                        *TRANSMIT AND RECEIVE FRAMES AT LOW SPEED IN INTERNAL
                        *LOOPBACK MODE WITHOUT INTERUPTIONS ON CHANEL A.
                        *
                        ****************************************************




                        ********************* TEST 6 *********************
                        *
                        *TRANSMIT AND RECEIVE FRAMES AT LOW SPEED IN INTERNAL
                        *LOOPBACK MODE WITHOUT INTERUPTIONS ON CHANEL B.
                        *
                        ****************************************************


                        ********************* TEST 7 *********************
                        *
                        *TRANSMIT AND RECEIVE FRAMES AT DIFFERNT SPEED IN
                        *INTERNAL LOOPBACK ON CHANAEL A  WITH INTERUPT
```

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




********************** TEST 8 *******************
*
*TRANSMIT AND RECEIVE FRAMES AT DIFFERNT SPEED IN
*INTERNAL LOOPBACK ON CHANAEL B WITH INTERUPT
*
**************************************************


NOTE:  TEST 9 TO THE END 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 12 *******************
*
*TEST MODEM SIGNAL CCITT 107 AND CCITT 108 ON
*CHANEL B  WITH EXTERNAL LOOPBACK
*(LOOP BACK CONNECTOR INSTALLED)
*
```

```
                        ;*****************************************************
                        ;
                        ;
                        ;
                        ;
                        ;
                        ;********************** TEST 13 ********************
                        ;*
                        ;*TEST MODEM SIGNAL CCITT 105,106,109 ON CHANEL A
                        ;* WITH EXTERNAL LOOPBACK
                        ;*(LOOP BACK CONNECTOR INSTALLED)
                        ;*
                        ;*****************************************************
                        ;
                        ;
                        ;
                        ;
                        ;********************** TEST 13 ********************
                        ;*
                        ;*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
- REVIEW ON DECEMBER 82

@

```
                                .TITLE KMV11 B LINE CNT  DIAG.
002000                                  .=2000
                                .MCALL  SVC
002000                          SVC                             ; INITIALIZE SUPERVISOR MACROS
                                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  VKMEA0,A,0,240.,0
```

```
1066
1067                                    ;++
1068                                    ; THIS TABLE IS USED BY THE RUNTIME SERVICES
1069                                    ; TO PROTECT THE LOAD MEDIA.
1070                                    ;--
1071
1072 002122                                    BGNPROT
1073
1074 002122  000000                            0               ;OFFSET INTO P-TABLE FOR CSR ADDRESS
1075 002124  177777                            -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
1076 002126  177777                            -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
1077
1078
1092
1093
1094 002130                                    ENDPROT
1095
```

```
1097                                    .SBTTL  DISPATCH TABLE
1098
1099                                    ;////////////////////////////////////////////////////////////////////////////////
1100                                    ;/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
1101                                    ;/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
1102                                    ;////////////////////////////////////////////////////////////////////////////////
1103
1104 002130                                    DISPATCH 12
1105
1112
1113
1114
1115
1116
```

DEFAULT HARDWARE P-TABLE

```
                                .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
002162                                  BGNHW   DFPTBL


002164  177000                          .WORD   177000          ;KMV11,CSRS ADDRESS
002166  000300                          .WORD   300             ;KMV11, VECTOR ADDRESS
002170  004000                          .WORD   4000            ;INTERRUPT PRIORITY LEVEL
002172  000001                          .WORD   1
002174                                  ENDHW
```

DEFAULT HARDWARE P-TABLE

```
    1147
    1148
    1149
    1150
    1151
    1152
```

```
                                .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
                                ;
                                ;
```

```
                                ; 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                          PWT==    1000
002000                          PRI==    2000
004000                          IXE==    4000
010000                          IBE==   10000
020000                          IER==   20000
040000                          LOE==   40000
100000                          HOE==  100000

1193
1194
1195    000340                  MAXPRI==340
1196    054000                  MAINT0==54000           ;MASTER CLEAR = 1,MODE = 1 ,MAINT 1 = 1 ,T11=HOLD
1197    044000                  MAINT1==44000           ;MASTER CLEAR = 1,MODE = 0 ,MAINT 1 = 0 ,T11=NOT HOLD
1198    040000                  MCLR==40000
1199    052525                  DATA1== 052525
1200    125252                  DATA2== 125252
1201    013224                  KB1.2== 5780.                       ;OCTAL VALUE OF 1.2 KBAUDS
1202    005473                  KB2.4==2875.                        ;  ""      ""   "" 2.4   ""
1203    000154                  KB64==  108.                        ;  ""      ""     64    ""
1204    000174                  KB56==  124.                        ;  ""      ""     56    ""
1205    000146                  KB68==  102.                        ;  ""      ""     68    ""
1206    000143                  KB70==  99.                         ;  ""      ""     70    ""
1207    000141                  KB72==  97.                         ;  ""      ""     72    ""
1208
1209                            ;                       DIVIDER CALCULATION= DECIMAL VALUE=6912 / X KBAUDS
1210
1211
1212
1213                            ;;*************************************************************************
1214                            ;* PROGRAM EVENT FLAG DEFINITIONS
1215                            ;;*************************************************************************
1216
1217
1218
```

```
                                .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                          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
```

```
1277                            ;*************************************************************************
1278                            ;* MISCELLANEOUS STORAGE
1279                            ;*************************************************************************
1280
1281 002266  000000            DELCT1: .WORD   0
1282 002270  000000            DELCT2: .WORD   0
1283 002272  000000            GOOD:   .WORD   0
1284 002274  000000            GOOD0:  .WORD   0
1285 002276  000000            GOOD1:  .WORD   0
1286 002300  000000            GOOD2:  .WORD   0
1287 002302  000000            GOOD4:  .WORD   0
1288 002304  000000            GOOD6:  .WORD   0
1289 002306  000000            GOOD10: .WORD   0
1290 002310  000000            GOOD12: .WORD   0
1291 002312  000000            GOOD14: .WORD   0
1292 002314  000000            GOOD16: .WORD   0
1293 002316  000000            SEL0:   .WORD   0
1294 002320  000000            SEL1:   .WORD   0
1295 002322  000000            SEL2:   .WORD   0
1296 002324  000000            SEL4:   .WORD   0
1297 002326  000000            SEL6:   .WORD   0
1298 002330  000000            SEL10:  .WORD   0
1299 002332  000000            SEL12:  .WORD   0
1300 002334  000000            SEL14:  .WORD   0
1301 002336  000000            SEL16:  .WORD   0
1302 002340  000000            BSEL1:  .WORD   0
1303 002342  000000            RANST:  .WORD   0
1304 002344  000000            RANSEL: .WORD   0
1305 002346  000000            RANMTA: .WORD   0
1306 002350  000000            RANDN:  .WORD   0
1307 002352  000000            SAVPC1: .WORD   0
1308 002354  000000            SAVSTA: .WORD   0
1309 002356  000000            COUNT:  .WORD   0
1310 002360  000000            NUMBER: .WORD   0
1311 002362  000000            ADDR:   .WORD   0
1312 002364  000000            GDDAT:  .WORD   0
1313 002366  000000            BDDAT:  .WORD   0
1314
1315 002370                     TTABLE: .BLKW   2000
1316 006370                     RTABLE: .BLKW   2000
1317
1318 012370  000000            EXADDR: .WORD   0
1319 012372  000000            INTFLG: .WORD   0
1320 012374  000000            BAD:    .WORD   0
1321 012376  000000            BSEL0:  .WORD   0
1322 012400  000000            DATA:   .WORD   0
1323 012402  000000            VECT:   .WORD   0
1324
1325
1326 012404  000000            KIND:   .WORD   0                       ;=0 IF KMV11A ,=1 IF KMV11B
1327 012406  000000            CHANEL: .WORD   0
1328
1329 012410  000000            TXDATA: .WORD   0
1330 012412  000000            RXDATA: .WORD   0
1331 012414  000000            TSPEED: .WORD   0
1332 012416  000000            LENGTH: .WORD   0
1333 012420  000000            NUB:    .WORD   0
```

GLOBAL DATA SECTION

```
   1334 012422  000000                RXCNT:  .WORD   0
   1335 012424  000000                STAERR: .WORD   0
   1336 012426  000000                WRDCNT: .WORD   0
```

```
1338
1339                            ;*****************************************************************
1340                            ;LOAD IN LOCATION "GDREV" THE PROM VERSION NUMBER THAT IS       *
1341                            ;COMPATIBLE WhITH THIS DIAGNOSTIC                              *
1342                            ;                                                              *
1343                            ;EACH PROM CONTAIN A REV LEVEL AND A ECO LEVEL:                *
1344                            ;THE REV LEVEL IS MODIFIED EACH TIME A MODIFICATION IS DONE    *
1345                            ;THE ECO LEVEL IS MODIFIED WHEN THE PROM MODIFICATION NEED     *
1346                            ;A DIAGNOSTIC MODIFICATION                                     *
1347                            ;*****************************************************************
1348
1349
1350
1351
1352
1353 012430  000001             GDREV:  .WORD   1
1354
1355
1356
1357
1358
```

```
1360                                ;*************************************************************************
1361                                ;* PROGRAM CONTROL FLAGS
1362                                ;*************************************************************************
1363 012432     000                INIFLG: .BYTE   0               ;PROGRAM INITIALIZING FLAG
1364                                        .EVEN
1365 012434     000                LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
1366 012435     000                QV.FLG: .BYTE   0               ;QUICK VERIFY FLAG
1367                                        .EVEN
1368 012436  000000                UUT:    .WORD   0               ;CURRENT UNIT UNDER TEST
1369
1370
1371
1372
1373
1374                                ;*************************************************************************
1375                                ;* POINTERS TO KMV11 VECTORS AND REGISTERS
1376                                ;*************************************************************************
1377 012440  000000                KMVV00: 0       ;POINTER TO KMV11  INTRPT VECTOR 0
1378 012442  000000                KMVLVL: 0       ;POINTER TO KMV11  INTRPT SERVICE
1379 012444  000000                KMVV04: 0       ;POINTER TO KMV11  INTRPT VECTOR 04
1380 012446  000000                KMVV02: 0       ;    ''       ''       ''       ''       02
1381 012450  000000                KMVV06: 0       ;    ''       ''       ''       :        06
1382 012452  000000                KMTLVL: 0       ;POINTER TO KMV11 TX INTRPT SERVICE PS
1383 012454  000000                KMVCSR: 0       ;POINTER TO KMV11 CONTROL STATUS REGISTER
1384 012456  000000                KMVP02: 0       ;POINTER TO KMV11 PORT REGISTER - SEL2
1385 012460  000000                KMVP04: 0       ;POINTER TO KMV11 PORT REGISTER - SEL4
1386 012462  000000                KMVP06: 0       ;POINTER TO KMV11 PORT REGISTER - SEL6
1387
1388 012464  000000                KMVP10: 0       ;POINTER TO KMV11 PORT REG  -SEL10
1389 012466  000000                KMVP12: 0       ;POINTER TO PORT REG  -SEL 14
1390 012470  000000                KMVP14: 0       ;POINTER TO PORT REG  -SEL14
1391 012472  000000                KMVP16: 0       ;POINTER TO PORT REG 16
1392
1393 012474  000000                LOOP:   0       ;POINTER TO LOOP BACK CONNECTOR
1394
1395
1396                                ;;****  PRIMARY REG ADRS STORAGE FOR THIS UNIT  ******
1397                                ;THESE LOCATIONS WILL BE LOADED FOR THE CURRENT UNIT, IN INIT CODE
1398 012476                        REGADR:
1399
1400                                ;;****  STACK USED FOR SUBROUTINE LINKAGE  *****
1401 012476                                .BLKW   100
1402 012676                        SSTACK:
1403
1404
1405
1406
1407
1408
1409
```

```
   1411                                  .SBTTL  GLOBAL TEXT SECTION
   1412
   1413                                  ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   1414                                  ;%      THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
   1415                                  ;%      MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
   1416                                  ;%      MORE THAN ONE TEST.
   1417                                  ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   1418
   1419                                  ;;****************************************************************************
   1420                                  ;* NAMES OF DEVICES SUPPORTED BY PROGRAM
   1421                                  ;;****************************************************************************
   1422 012676                                  DEVTYP  <KMV11B>
   1423
   1424
   1425                                  ;
   1426                                  ; FORMAT STATEMENTS USED IN PRINT CALLS
   1427                                  ;
   1428
   1435
   1436
   1437
   1438
   1439
```

```
1441                                    .SBTTL  GLOBAL SUBROUTINES
1442
1443
1444
1445
1446
1447                                    ;-----------------------------------------------------------------
1448                                    ; MACRO'S NEEDED TO CALL SUBROUTINES
1449                                    ;-----------------------------------------------------------------
1450
1451                                    .MACRO  CLRMAR
1452                                            ROMCLK
1453                                            004000
1454                                    .ENDM   CLRMAR
```

```
                                ;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
```

```
1497                                    ;MACRO TO WAIT A FEW MS
1498
1499
1500                                    ;CALLING SEQUENCE:      WAITA   X               0<X<177777
1501                                    ;                       WAITB   X,Y             0<X OR Y<177777
1502
1503
1504
1505                                    .MACRO  WAITA   X
1506                                            MOV     #X,DELCT1               ;LOAD COUNT
1507                                            JSR     PC,WAIT1                ;WAIT
1508                                    .ENDM
1509
1510
1511
1512
1513
1514
1515
1516                                    .MACRO  WAITB   X,Y
1517                                            MOV     #X,DELCT1
1518                                            MOV     #Y,DELCT2
1519                                            JSR     PC,WAIT2
1520                                    .ENDM
1521
1522
```

```
1524
1525                                    ;ROUTINE TO DROP UNIT AFTER 5 ERROR
1526
1527
1528                                    ;JSR   PC,CHKMAX
1529
1530
1531
1532
1533
1534
1535
1536
1537 012740                            CHKMAX: INLOOP                          ;LOOPING ON ERROR?
1538 012742                                    BCOMPLETE       1$              ;IF YES, EXIT
1539
1540
1541 012744                                    RFLAGS  R0                      ;GET OPERATOR FLAG
1542 012746  032700  000040                    BIT     #IDU,R0                 ;IS DROPPING INHIBITTED?
1543 012752  001026                            BNE     1$                      ;IF YES EXIT
1544
1545
1546 012754  005237  002236                    INC     ERRCNT                  ;UPDATE ERROR COUNT
1547 012760  023737  002236  002234           CMP     ERRCNT,MAXERR           ;TOO MANY ERROR?
1548 012766  003420                            BLE     1$                      ;IF NOT JUMP
1549
1550
1551 012770                                    PRINTF  #NERRS,MAXERR,UUT       ;TOO MANY ERROR!
1552 013020                                    DODU    UUT                     ;DROP UNIT
1553
1554 013026                                    DOCLN                           ;END THE SUBPASS
1555
1556 013030  000207                    1$:     RTS     PC
1557
1558
1559
1560
1561                                           .NLIST  BEX
1562 013032      045      116      045  NERRS: .ASCIZ  /%N%AMORE THAN %D3%A  ERRORS ON UNIT %D2/
1563                                           .LIST   BEX
1564                                           .EVEN
1565
1566
1567
1568
1569
```

```
                                ;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
```

```
1626                    .SBTTL  NUMBER GENERATOR
1627
1628
1629
1630                    DESCRIPTION:
1631
1632                            ROUTINE TO GENERATE DATA PATTERNS,
1633                    THE TYPE OF PATTERN IS SELECTED BY R3, AND THE
1634                    PATTERN GENERATED IS RETURNED IN LOCATION "DATA"
1635                    AND LOCATION "GOOD"
1636
1637                    CALLING SEQUENCE:
1638
1639                            JSR     PC,GENER
1640
1641                    INPUT PARAMETERS:
1642
1643                    R3 CONTAINS THE PATTERN NUMBER
1644
1645                    R3=0            ALL ZEROES
1646                    1               ALL ONES
1647                    2               010101 ETC BIT PATTERN
1648                    3               101010 ETC BIT PATTERN
1649                    4               ROTATING 1 IN A ZERO WORD
1650                    5               ROTATING 0 IN AN ALL ONE WORD
1651                    6               PSEUDO RANDOM NUMBER
1652                    7               INCREMENTING DATA PATTERN, GOOD
1653                                    CONTAINS THE VALUE TO BE UPDATED
1654
1655
1656
1657
1658                    IMPLICIT INPUT PARAMETERS:
1659
1660                            NONE
1661
1662
1663                    OUTPUT PARAMETERS:
1664
1665                            THE NUMBER GENERATED IS HELD IN
1666                    DATA AND GOOD.
1667
1668
1669                    IMPLICIT OUTPUT PARAMETERS:
1670
1671                            NONE
1672
1673
1674                    COMPLETION CODES:
1675
1676                            NONE
1677
1678
1679                    POSSIBLE ERROR CODES:
1680
1681                            NONE
1682
```

```
1683                                    ;
1684                                    ;
1685 013160  042703  177770            GENER:  BIC     #177770,R3
1686 013164  004737  013460                    JSR     PC,SAVREG
1687 013170  006303                            ASL     R3
1688 013172  000173  013176                    JMP     @GENSEL(R3)
1689 013176  013216                    GENSEL: GEN0                    ;ALL ZERO WORD
1690 013200  013222                            GEN1                    ;ALL ONE WORD
1691 013202  013230                            GEN52                   ;52 PATTERN
1692 013204  013236                            GEN25                   ;25 PATTERN
1693 013206  013244                            GENR1                   ;ROTATE '1' EACH CALL
1694 013210  013254                            GENR0                   ;ROTATE '0' EACH CALL
1695 013212  013312                            GENRAN                  ;RANDOM NUMBER
1696 013214  013432                            GENINC                  ;INCREMENTING COUNT
1697 013216  005000                    GEN0:   CLR     R0              ;0>R0
1698 013220  000507                            BR      GENEX
1699 013222  005000                    GEN1:   CLR     R0              ;NOT0>R0
1700 013224  005100                            COM     R0
1701 013226  000504                            BR      GENEX
1702 013230  012700  052525            GEN52:  MOV     #52525,R0       ;5252>R0
1703 013234  000501                            BR      GENEX
1704 013236  012700  125252            GEN25:  MOV     #125252,R0      ;125252>R0
1705 013242  000476                            BR      GENEX
1706 013244  000241                    GENR1:  CLC
1707 013246  004737  013266                    JSR     PC,GENROT       ;SHIFT 1 > R0
1708 013252  000472                            BR      GENEX
1709 013254  000241                    GENR0:  CLC
1710 013256  004737  013266                    JSR     PC,GENROT       ;
1711 013262  005100                            COM     R0              ;SHIFT 0 > R0
1712 013264  000465                            BR      GENEX
1713 013266  006037  013310            GENROT: ROR     GENISH          ;ROTATE  1 PATTERN
1714 013272  001003                            BNE     GENER1          ;= 0?
1715 013274  012737  100000  013310            MOV     #100000,GENISH  ;YES, SET MSB
1716 013302  013700  013310            GENER1: MOV     GENISH,R0       ;PUT 1 IN R0
1717 013306  000207                            RTS     PC              ;AND EXIT
1718 013310  000001                    GENISH: 1
1719 013312  012737  000005  002344    GENRAN: MOV     #5,RANSEL       ;SET SELECT VALUE TO 5
1720 013320  004737  013332                    JSR     PC,RANGEN       ;GENERATE RANDOM NUMBER IN R0
1721 013324  013700  002350                    MOV     RANDN,R0        ;
1722 013330  000443                            BR      GENEX           ;
1723 013332  013702  002350            RANGEN: MOV     RANDN,R2        ;
1724 013336  001002                            BNE     RAN1            ;IS RANDOM = 0
1725 013340  013702  002342                    MOV     RANST,R2        ;YES, PUT RANDOM START VALUE IN
1726 013344  032737  000777  002344    RAN1:   BIT     #777,RANSEL     ;NO;IS RANSEL SELECT VALUE = 0
1727 013352  001003                            BNE     RAN2            ;NO
1728 013354  012737  000001  002344            MOV     #1,RANSEL       ;YES: SET RANSEL = 1
1729 013362  013703  002344            RAN2:   MOV     RANSEL,R3       ;
1730 013366  013702  002350                    MOV     RANDN,R2
1731 013372  033702  002346                    BIT     RANMTA,R2       ;GET R2 <0 AND 1>
1732 013376  001405                            BEQ     RANCLC          ;
1733 013400  005102                            COM     R2
1734 013402  033702  002346                    BIT     RANMTA,R2
1735 013406  001401                            BEQ     RANCLC
1736 013410  000402                            BR      RANSEC
1737 013412  000241                    RANCLC: CLC
1738 013414  000401                            BR      RAN4
1739 013416  000261                    RANSEC: SEC
```

```
1740 013420  006037  002350          RAN4:   ROR     RANDN           ;ROTATE C TO B15
1741 013424  005303                          DEC     R3              ;IS THIS NUMBER REQUIRED?
1742 013426  001357                          BNE     RAN2+4          ;NO, GET ANOTHER
1743 013430  000207                  RANEX:  RTS     PC              ;YES, EXIT
1744 013432  013700  002272          GENINC: MOV     GOOD,R0         ;INCREMENTS LOC. 'GOOD'
1745 013436  005200                          INC     R0
1746 013440  010037  002272          GENEX:  MOV     R0,GOOD
1747 013444  004737  013540                  JSR     PC,RSTREG
1748 013450  013737  002272  012400          MOV     GOOD,DATA
1749 013456  000207                          RTS     PC
1750
```

```
                        .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

```
1809 013500  012637  002352                MOV     (SP)+,SAVPC1
1810 013504  010546                        MOV     R5,-(SP)
1811 013506  010446                        MOV     R4,-(SP)
1812 013510  010346                        MOV     R3,-(SP)
1813 013512  010246                        MOV     R2,-(SP)
1814 013514  010146                        MOV     R1,-(SP)
1815 013516  010046                        MOV     R0,-(SP)
1816 013520  013746  002352                MOV     SAVPC1,-(SP)
1817 013524  013746  002246                MOV     SAVPC,-(SP)     ;PUT PC READY FOR
1818 013530                                SETPRI  SAVSTA
1819 013536  000207                        RTS     PC              ;RETURN
1820
1821
1822
```

```
                                .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

```
1881 013570  012602                        MOV     (SP)+,R2
1882 013572  012603                        MOV     (SP)+,R3
1883 013574  012604                        MOV     (SP)+,R4
1884 013576  012605                        MOV     (SP)+,R5
1885 013600  013746  002352                MOV     SAVPC1,-(SP)
1886 013604  013746  002246                MOV     SAVPC,-(SP)     ;PUT PC READY FOR
1887 013610                                SETPRI  SAVSTA
1888 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          CKSELO: MOV     (R5)+,GOOD      ;WRITE GOOD
013624  017737  176624  002316          MOV     @KMVCSR,SELO    ;READ SEL 0
013632  023737  002316  002272          CMP     SELO,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          CBSELO: CLR     GOOD
013656  012537  002272                  MOV     (R5)+,GOOD
013662  117737  176566  012376          MOVB    @KMVCSR,BSELO
013670  123737  012376  0022?2          CMPB    BSELO,GOOD
013676  001001                          BNE     1$
013700  000402                          BR      2$
013702  062705  000002          1$:     ADD     #2,R5
013706  000205                  2$:     RTS     R5
```

```
                                ;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

```
1984 014124  001011                          BNE     1$
1985 014126  023737  002334  002312          CMP     SEL14,GOOD14
1986 014134  001005                          BNE     1$
1987 014136  023737  002336  002314          CMP     SEL16,GOOD16
1988 014144  001001                          BNE     1$
1989
1990 014146  000402                          BR      2$
1991 014150  062705  000002          1$:     ADD     #2,R5
1992 014154  000205                  2$:     RTS     R5
```

```
                                        ;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
```

```
                                        ;ROUTINE TO CLEAR KMV11 MODULE


                                        ;CALLING SEQUENCE:
                                        ;       JSR PC,CLRKMV

                                        ;ROUTINE DESCRIPTION: CLEAR ALL CSR'S REGISTERS AND CHECK IF = 0


014400                                  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
```

```
                                        ;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
```

```
                                ;ROUTINE TO SET TEST NUMBER ON BSEL0
                                ;
                                ;
                                ;
                                ;CALLING SEQUENCE:
                                ;       JSR R5,TSTNUB
                                ;       .WORD   A                       A=TEST MICRO PROGRAM NUMBER
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
014552  012537  012420          TSTNUB: MOV     (R5)+,NUB
014556  053777  012420  175670          BIS     NUB,@KMVCSR             ;LOAD TEST NUMBER
014564  012737  000000  002266          MOV     #0000,DELCT1
014572  004737  012726                  JSR     PC,WAIT1                ;WAIT
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    ;WRITTE LOOP BIT
014656  042737  177776  012474          BIC     #177776,LOOP
014664  017737  175570  012404          MOV     @KMVP04,KIND    ;WRITTE KIND
014672  042737  177776  012404          BIC     #177776,KIND
```

RESTORE REGISTERS

```
   2192 014700  000207                          RTS     PC
```

```
                                ;ROUTIN TO WRITE OR READ ONE OF THE KMV11 REGISTERS



                                ;CALLING SEQUENCE:
                                ;JSR     R5,WRITE
                                ;.WORD   A                              A=ADDRESS TO WRITE
                                ;.WORD   B                              B=DATA TO WRITE
                                ;
                                ;
                                ;
                                ;JSR     R5,READ
                                ;.WORD   A                              A=ADDRESS TO READ
                                ;
                                ;
                                ;MICRO DIAG NB 47 DESCRIPTION:
                                ;WRITE: PUT ADDRESS TO WRITE IN SEL2
                                ;       PUT DATA TO WRITE IN SEL4
                                ;       SET BIT 0 OF SEL6(WRITE BIT)
                                ;       SET TEST NB 44
                                ;       KMV11 CLEAR BSEL0 WHEN DONE
                                ;
                                ;READ:  PUT ADDRESS TO READ IN SEL2
                                ;       CLEAR BIT 0 IN SEL6
                                ;       SET TEST 47
                                ;       KMV11 READ ADDRESS IN SEL2 AND CLEAR BSEL0 WHEN DONE


014702  012577  175550          WRITE:  MOV     (R5)+,@KMVP02           ;WRITE ADDRESS
014706  012577  175546                  MOV     (R5)+,@KMVP04           ;   "    DATA
014712  012777  000001  175542          MOV     #1,@KMVP06              ;BIT WRITE

014720  004537  014552                  JSR     R5,TSTNUB               ;SEND TEST NB 47
014724  000047                          .WORD   47

014726  000205                          RTS     R5                      ;RETURN



014730  012577  175522          READ:   MOV     (R5)+,@KMVP02           ;SET ADDRESS TO READ
014734  005077  175520                  CLR     @KMVP04
014740  005077  175516                  CLR     @KMVP06

014744  004537  014552                  JSR     R5,TSTNUB               ;SEND TEST NB 44
014750  000047                          .WORD   47

014752  004737  013102                  JSR     PC,TSTERR               ;CHECK BSEL 0
014756  000412                          BR      1$                      ;OK
```

```
2251 014760  000402                          BR      2$
2252 014762  000401                          BR      2$
2253 014764  000400                          BR      2$
2254
2255 014766                          2$:     ERRHRD  4,EM0004              ;NO KMV ANSWER
2256 014776  004737  012740                  JSR     PC,CHKMAX             ;MAX ERROR?
2257 015002  000205                          RTS     R5
2258
2259 015004  017737  175450  012374  1$:     MOV     @KMVP04,BAD           ;READ DATA IN BAD
2260 015012  000205                          RTS     R5
2261
2262
2263
```

```
2265
2266
2267                            .MACRO  ROMCLK
2268                            .LIST
2269                                    JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
2270                            .NLIST
2271                            .ENDM
2272
2273                            .MACRO  ED$CALL XY
2274                                    .LIST
2275                            ;********************** TEST'XY' **************
2276                                    .NLIST
2277                                    .ENDM
2278
2279
2280
2281                            .MACRO  BADHEAD
2282                                    .RADIX  10
2283                                    ED$CALL \T$TESTNUM+1
2284                                    .RADIX  8
2285                                    .ENDM
2286
2287
2288
2289
```

```
                                        .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 FAIL TO RESET: DCT11 CAN'T CLEAR MASTER CLEAR /

     015214     040     113     115     EM0002: .ASCIZ  / KMV11 REGISTERS CAN'T BE CLEARED /

     015257     040     104     101     EM0003: .ASCIZ  / DATA COMPARE ERROR ON KMV11 REGISTER (SEL2 TO SEL16)/

     015345     040     116     117     EM0004: .ASCIZ  / NO ANSWER FROM KMV11 /

     015374     040     124     111     EM0006: .ASCIZ  / TIMEOUT DURING KMV11 MICRO TEST /

     015436     111     116     124     EM0007: .ASCIZ  /INTERUPT OCCURED ON KMV11,WHEN ADDRESSING WRONG CSR REGISTER/

     015533     113     115     126     EM0010: .ASCIZ  /KMV11 DON'T RECEIVE ANY INTERUPT WHEN QBUS ADDRESS CSR'S /

     015625     113     115     126     EM0011: .ASCIZ  /KMV11 REAL LINE TIME CLOCK FAILED TO INTERUPT /

     015704     103     110     101     EM0012: .ASCIZ  /CHANEL A   GENERATOR COUNT CAN'T BE READ OR WRITTE CORRECTLY /

     016002     103     110     101     EM0013: .ASCIZ  /CHANEL A  GENERATOR OUTPUT ISN'T IN A GOOD STATE/

     016063     103     110     101     EM0112: .ASCIZ  /CHANEL B   GENERATOR COUNT CAN'T BE READ OR WRITTE CORRECTLY /

     016161     103     110     101     EM0113: .ASCI   /CHANEL B  GENERATOR OUTPUT ISN'T IN A GOOD STATE/

     016242     116     117     040     EM0033: .ASCIZ  /NO CHANGE IN BAUD RATE GENERATOR COUNT /

     016312     116     117     040     EM0014: .ASCIZ  /NO ACTION ON BAUD RATE GENERATOR OUTPUT /

     016363     105     122     122     EM0015: .ASCIZ  /ERROR WHEN TRANSMITTING ON CH A IN INTERNAL LOOP NO INTERUPTS/

     016461     105     122     122     EM0016: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN INTERNAL LOOPBACK ON CH A /

     016556     105     122     122     EM0115: .ASCIZ  /ERROR WHEN TRANSMITTING ON CH B IN INTERNAL LOOP NO INTERUPTS/

     016654     105     122     122     EM0116: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN INTERNAL LOOPBACK ON CH B /

     016751     105     122     122     EM0017: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN EXTERNAL LOOP BACK ON CH A/

     017046     105     122     122     EM0020: .ASCIZ  /ERROR WHEN TRANSMITTING FRAMES IN EXTERNAL LOOP BACK ON CH B/

     017143     122     105     101     EM0023: .ASCIZ  /REAL TIME CLOCK INTERUPT OCCURED TOO EARLY /

     017217     111     116     103     EM0024: .ASCIZ  /INCORRECT KMV11 REPLY /

```

```
2348 017246      116      117      040  EM0027: .ASCIZ  /NO LOOP BACK CONNECTOR ,TEST NOT EXECUTED/
2349
2350 017320      104      101      124  EM0030: .ASCIZ  /DATA COMPARE ERROR BETWEEN MODEM SIGNAL 108 AND 107 ON CH A /
2351
2352 017415      104      101      124  EM0130: .ASCIZ  /DATA COMPARE ERROR BETWEEN MODEM SIGNAL 108 AND 107 ON CH B /
2353
2354 017512      115      117      104  EM0032: .ASCIZ  /MODEM SIGNAL ERROR ON CHANEL A IN EXTERNAL LOOPBACK MODE/
2355
2356 017603      115      117      104  EM0034: .ASCIZ  /MODEM SIGNAL ERROR ON CHANEL B IN EXTERNAL LOOPBACK MODE/
2357
2358 017674      120      122      117  EM0035: .ASCIZ  /PROM REVISION IS NOT COMPATIBLE  /
2359
2360
2361
2362
```

```
2364 017736     045     116     045  MLOOP:  .ASCIZ  /%N%A   NO LOOP BACK CONNECTOR,TEST NOT EXECUTED/
2365
2366 020014     045     116     045  MSEL0:  .ASCIZ  /%N%A SEL0 = %06%A   SHOULD BE = %06%N/
2367
2368 020062     045     116     045  MREG0:  .ASCIZ  /%N%A SEL0 = %06%A   SHOULD BE = %06/
2369 020126     045     116     045  MREG2:  .ASCIZ  /%N%A SEL2 = %06%A   SHOULD BE = %06/
2370 020172     045     116     045  MREG4:  .ASCIZ  /%N%A SEL4 = %06%A   SHOULD BE = %06/
2371 020236     045     116     045  MREG6:  .ASCIZ  /%N%A SEL6 = %06%A   SHOULD BE = %06/
2372 020302     045     116     045  MREG10: .ASCIZ  /%N%A SEL10= %06%A   SHOULD BE = %06/
2373 020346     045     116     045  MREG12: .ASCIZ  /%N%A SEL12= %06%A   SHOULD BE = %06/
2374 020412     045     116     045  MREG14: .ASCIZ  /%N%A SEL14= %06%A   SHOULD BE = %06/
2375 020456     045     116     045  MREG16: .ASCIZ  /%N%A SEL16= %06%A   SHOULD BE = %06/
2376
2377
2378
2379 020522     045     116     045  MINT:   .ASCIZ  /%N%A GOOD = %06%A  BAD =%06/
2380 020556     045     116     045  MBSEL0: .ASCIZ  /%N%A BSEL0 =%06%A  SHOULD BE =%06/
2381
2382 020620     045     116     045  MINTR:  .ASCIZ  /%N%A INTERUPT OCCUR  WHEN ADDRESS =%06%A IS WRITTEN/
2383
2384 020704     045     116     045  MVECT:  .ASCIZ  /%N%A RECEIVE BAD VECT =%06%A SHOULD BE =%06/
2385
2386
2387
2388
2389
2390 020760     045     116     045  MT11V:  .ASCIZ  /%N%A   RECEIVE VECTOR VECT =%06%A SHOULD BE = %06/
2391 021042     045     116     045  MFRAM1: .ASCIZ  /%N%A   RECEIVE FRAME IS =%06%A SHOULD BE =%06/
2392 021120     045     116     045  MFRAM2: .ASCIZ  /%N%A   TRANSMIT SPEED IS  =%06%A FRAME LENGTH =%06/
2393
2394
2395 021203     045     116     045  MSTER1: .ASCIZ  /%N%A   ERROR STATUS  =%06/
2396 021235     045     116     045  MSTER2: .ASCIZ  /%N%A   WORD COUNT DISCREPANCY =%06/
2397
2398 021300     045     116     045  MODEM1: .ASCIZ  /%N%A   TESTED MODEM SIGNAL IS  =%06/
2399 021344     045     116     045  MODEM2: .ASCIZ  /%N%A   RESULT OF TEST IS  =%06/
2400 021403     045     116     045  MODEM3: .ASCIZ  /%N%A   MODEM SIGNAL STATE IS  =%06/
2401 021446     045     116     045  MRAMEF: .ASCIZ  /%N%A   TXDATA = %06%A      RXDATA = %06/
2402
```

```
                                        .EVEN



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

021516                                          BGNMSG  PRSELO                  ;REPORT SELO
021516                                          PRINTB  #MSELO,SELO,GOOD
021546  004737  012740                          JSR     PC,CHKMAX
021552                                          ENDMSG



021554                                  BGNMSG  PRINT
021554                                  PRINTB  #MINT,GOOD,BAD
021604  004737  012740                  JSR     PC,CHKMAX                       ;CHECK IF TOO MANY ERROR
021610                                  ENDMSG


021612                                  BGNMSG  PRALL                           ;REPORT CONTENT OF ALL CSR'S
021612                                  PRINTB  #MREG0,SEL0,GOOD0
021642                                  PRINTB  #MREG2,SEL2,GOOD2
021672                                  PRINTB  #MREG4,SEL4,GOOD4
021722                                  PRINTB  #MREG6,SEL6,GOOD6
021752                                  PRINTB  #MREG10,SEL10,GOOD10
022002                                  PRINTB  #MREG12,SEL12,GOOD12
022032                                  PRINTB  #MREG14,SEL14,GOOD14
022062                                  PRINTB  #MREG16,SEL16,GOOD16
022112  004737  012740                  JSR     PC,CHKMAX                       ;CHECK IF TOO MANY ERROR
022116                                  ENDMSG






022120                                  BGNMSG  PRREG                           ;REPORT ALL CSR'S BUT SELO
022120                                  PRINTB  #MREG2,SEL2,GOOD2
022150                                  PRINTB  #MREG4,SEL4,GOOD4
022200                                  PRINTB  #MREG6,SEL6,GOOD6
022230                                  PRINTB  #MREG10,SEL10,GOOD10
022260                                  PRINTB  #MREG12,SEL12,GOOD12
022310                                  PRINTB  #MREG14,SEL14,GOOD14
022340                                  PRINTB  #MREG16,SEL16,GOOD16
022370  004737  012740                  JSR     PC,CHKMAX                       ;CHECK IF TOO MANY ERROR
022374                                  ENDMSG







022376                                  BGNMSG  PBSELO                          ;REPORT BSELO
```

```
2461 022376                          PRINTB  #MBSEL0,BSEL0,GOOD
2462 022426  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2463 022432                          ENDMSG
2464
2465
2466
2467
2468
2469
2470
2471
2472 022434                          BGNMSG  PINTR                        ;REPORT INTERUPT
2473 022434                          PRINTB  #MINTR,ADDR
2474 022460  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2475 022464                          ENDMSG
2476
2477
2478
2479
2480
2481 022466                          BGNMSG  PVECT                        ;REPORT VECTOR
2482 022466                          PRINTB  #MVECT,VECT,GOOD
2483 022516  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2484 022522                          ENDMSG
2485
2486
2487
2488
2489 022524                          BGNMSG  PRT11V
2490 022524                          PRINTB  #MT11V,VECT,GOOD
2491 022554  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2492 022560                          ENDMSG
2493
2494
2495
2496
2497 022562                          BGNMSG  PFRAME                       ;REPORT FRAME ERROR
2498 022562                          PRINTB  #MFRAM1,RXDATA,TXDATA
2499 022612                          PRINTB  #MFRAM2,TSPEED,LENGTH
2500 022642  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2501 022646                          ENDMSG
2502
2503
2504
2505
2506
2507 022650                          BGNMSG  PMODEM                       ;REPORT MODEM SIGNAL  ERROR
2508 022650                          PRINTB  #MODEM1,GOOD
2509 022674                          PRINTB  #MODEM2,BAD
2510 022720                          PRINTB  #MODEM3,DATA
2511 022744  004737  012740          JSR     PC,CHKMAX                    ;CHECK IF TOO MANY ERROR
2512 022750                          ENDMSG
2513
2514
2515
2516
2517
```

```
2518
2519
2520 022752                            BGNMSG  PRAMEF
2521 022752                            PRINTB  #MRAMEF,TXDATA,RXDATA          ;SHORT REPORT FOR FRAME ERROR
2522 023002                            ENDMSG
2523
2524
2525
2526
2527
2528
2529
2530
2531 023004                            BGNMSG  PRSTER                                 ;REPORT ERROR STATUS ,WORD CNT
2532 023004                            PRINTB  #MSTER1,STAERR
2533 023030                            PRINTB  #MSTER2,WRDCNT
2534 023054  004737  012740            JSR     PC,CHKMAX                      ;CHECK IF TOO MANY ERROR
2535 023060                            ENDMSG
2536
2537
2538
2539 023062                            BGNMSG  PADFLT                         ;ADDRESS TEST
2540 023062                            PRINTB  #TFM36,ADDR,UNIT
2541 023112  004737  012740            JSR     PC,CHKMAX
2542 023116                            ENDMSG
2543
2544
2545
2546
```

```
                                .SBTTL  REPORT CODING SECTION


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

023120                                  BGNRPT


023120                                  EXIT    RPT


023124                                  ENDRPT
```

```
                                .SBTTL  INITIALIZE SECTION

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

023126                                  BGNINIT



                                        .EVEN

                                        .EVEN



                                ;INITIALIZE SUBROUTINE STACK
023126  012705  01267'                  MOV     #SSTACK,R5
                                ;STORE BASE LEVEL PROGRAM STACK POINTER
023132  010637  002242                  MOV     SP,PSTACK
023136  005737  002252                  TST     FTIME
023142  001011                          BNE     1$
023144  013737  000004  002254          MOV     @#4,SAVE4
023152  013737  000006  002256          MOV     @#6,SAVE6
023160  012737  000001  002252          MOV     #1,FTIME
023166  013737  002254  000004  1$:     MOV     SAVE4,@#4
023174  013737  002256  000006          MOV     SAVE6,@#6

023202                                  READEF  #EF.START                       ;START COMMAND?
023210                                  BCOMPLETE       SETUP                   ;IF YES BRANCH

023212                                  READEF  #EF.CONTINUE                    ;CONTINUE COMMAND?
023220                                  BCOMPLETE       END

023222                                  READEF  #EF.NEW                         ;NEW PASS?
023230                                  BNCOMPLETE      NEXT                    ;IF NOT EXIT SETUP

023232  012737  177777  012436  SETUP:  MOV     #-1,UUT                         ;INITIALISE UNIT NUMBER

023240  005237  012436          NEXT:   INC     UUT                             ;POINT NEXT UNIT
023244  023737  012436  002262          CMP     UUT,L$UIT                       ;ALL DONE?
023252  001523                          BEQ     ABORT                           ;IF YES END OF PASS

023254  013737  012436  002264          MOV     UUT,UNIT
023262                                  PRINTF  #RUNNING,UNIT


023306                                  GPHARD  UUT,R1                          ;GET P TABLE
023316                                  BNCOMPLETE      NEXT                    ;IF NOT AVAILABLE GET NEXT


023320                          GETPRM:                         ;GET ADDRESS OF KMV11

023320  011137  012454                  MOV     (R1),KMVCSR
                                                                ;GET POINTER TO KMV11 SEL02 REG
023324  011137  012456                  MOV     (R1),KMVP02
```

```
2668 023330  062737  000002  012456          ADD     #2,KMVP02
2669                                                                ;GET POINTER TO KMV11 PORT REG - SEL 4
2670 023336  011137  012460                  MOV     (R1),KMVP04
2671 023342  062737  000004  012460          ADD     #4,KMVP04
2672                                                                ;GET POINTER TO KMV11 PORT REG - SEL 6
2673 023350  011137  012462                  MOV     (R1),KMVP06
2674 023354  062737  000006  012462          ADD     #6,KMVP06
2675                                                                ;GET POINTER TO KMV11 REG 10
2676 023362  011137  012464                  MOV     (R1),KMVP10
2677 023366  062737  000010  012464          ADD     #10,KMVP10
2678                                                                ;GET POINTER TO KMV11 REG 12
2679 023374  011137  012466                  MOV     (R1),KMVP12
2680 023400  062737  000012  012466          ADD     #12,KMVP12
2681                                                                ;GET POINTER TO KMV11 REG 14
2682 023406  011137  012470                  MOV     (R1),KMVP14
2683 023412  062737  000014  012470          ADD     #14,KMVP14
2684                                                                ;GET POINTER TO KMV11 REG 16
2685 023420  012137  012472                  MOV     (R1)+,KMVP16
2686 023424  062737  000016  012472          ADD     #16,KMVP16
2687                                                                ;GET POINTER TO  VECTOR 0
2688 023432  011137  012440                  MOV     (R1),KMVV00
2689                                                                ;GET POINTER TO  VECTOR 2
2690 023436  011137  012446                  MOV     (R1),KMVV02
2691 023442  062737  000002  012446          ADD     #2,KMVV02
2692                                                                ;GET POINTER TO VECTOR 4
2693 023450  011137  012444                  MOV     (R1),KMVV04
2694 023454  062737  000004  012444          ADD     #4,KMVV04
2695                                                                ;GET POINTER TO VECTOR 6
2696 023462  012137  012450                  MOV     (R1)+,KMVV06
2697 023466  062737  000006  012450          ADD     #6,KMVV06
2698                                                                ;GET POINTER TO TX PRIORITY LEVEL
2699 023474  012137  012442                  MOV     (R1)+,KMVLVL
2700 023500  062737  000006  012452          ADD     #6,KMTLVL
2701                                                                ;GET LOOPBACK PARAMETERS:
2702 023506  011137  012474                  MOV     (R1),LOOP
2703
2704 023512  005037  002236                  CLR     ERRCNT          ;CLEAR ERROR COUNT
2705 023516                                  EXIT    INIT
2706
2707
2708
2709 023522                          ABORT:  DOCLN                           ;CLEAN UP AND ABORT PASS
2710 023524                                  EXIT INIT                       ;EXIT
2711
2712
2713 023530                          END:    ENDINIT
2714
2715
2716                                         .NLIST  BEX
2717 023532      045     116     045 RUNNING:        .ASCIZ  /%N%A RUNNING ON UNIT %D2%A  PASS TIME=3,5 MINUTES/
2718                                         .LIST   BEX
2719                                         .EVEN
2720
2721
2722
```

```
                                        .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
```

```
                                .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
```

```
                                        BADHEAD
                                ;********************** TEST1 **************
                                        ;*VERIFY THAT REFERENCING UNIBUS DEVICE REGISTERS
                                        ;*DOES NOT CAUSE A TIME OUT TRAP
                                        BADHEAD
                                ;********************** TEST1 **************

                                BGNTST
                                        MOV     KMVCSR,R1       ;R1 CONTAINS KMV11 ADDRESSES
                                        MOV     #7,R5           ;7 REGISTERS TO BE TESTED
                                        MOV     #2$,4           ;SET OUT TIMEOUT TRAP
                                        MOV     #340,6          ;LEVEL 7
                                1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
                                        NOP
                                        ESCAPE  TST
                                        ADD     #2,R1           ;NEXT REGISTER
                                        DEC     R5              ;DEC REGISTER COUNT
                                        BNE     1$              ;BP IF NOT LAST REGISTER
                                        BR      3$

                                2$:     ADD     #4,SP
                                        MOV     R1,ADDR
                                        MOV     UUT,UNIT
                                        ERRHRD  0,TIM,PADFLT
                                        ESCAPE  TST

                                3$:     MOV     SAVE4,4
                                        MOV     SAVE6,6
                                        ESCAPE  TST

                                ENDTST
                                .EVEN
```

```
2980 024116                          BADHEAD
                                     ;********************** TEST2 **************
2981                                 ;CHECK PROM REVISION TO SEE IF COMPATIBLE WHITH DIAGNOSTIC
2982 024116                          BADHEAD
                                     ;********************** TEST2 **************
2983
2984
2985
2986
2987 024116                          STARS 1
2988                                 ;READ LOCATION 2 OF THE PROM (ADDRESS 160002) WHICH CONTAINS PROM VERSION
2989                                 ;       NUMBER
2990                                 ;CHECK IF DIAGNOSTIC AND PROM ARE COMPATIBLE AND GIVE AN ERROR IF NOT
2991 024116                          STARS 1
2992
2993
2994
2995
2996
2997 024116                          BGNTST
2998 024116  004737  014400                  JSR     PC,CLRKMV               ;CLEAR ALL REGISTERS
2999 024122  004737  014502                  JSR     PC,MAINM1               ;SET MAINT MODE
3000
3001
3002 024126  004537  014730          REVPRO: JSR     R5,READ                 ;READ LOCATION 160002
3003 024132  160002                          .WORD   160002
3004
3005
3006 024134  023737  012430  012374        CMP     GDREV,BAD               ;LOOK IF COMPATIBLE
3007 024142  001410                          BEQ     1$                      ;YES
3008
3009 024144                                  ERRHRD  7,EM0035                ;REPORT THE ERROR
3010 024154  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3011 024160                                  ESCAPE TST
3012 024164                          1$:
3013 024164                          ENDTST
```

```
3015
3016
3017 024166                             BADHEAD
                                        ;********************** TEST3 **************
3018                                    ;REAL TIME CLOCK TEST
3019 024166                             BADHEAD
                                        ;********************** TEST3 **************
3020
3021
3022
3023
3024
3025
3026 024166                             STARS 1
3027                                    ;TEST DESCRIPTION:
3028                                    ;THIS TEST CHECK KMV11 REAL TIME CLOCK.
3029                                    ;THE DCT11 FULLY EXECUTE THIS MICRO TEST AND GIVE A TEST RESULT
3030                                    ;VIA CSR'S TO THE HOST COMPUTER.( TIMING IS CHECKED BY DCT11)
3031                                    ;
3032                                    ;
3033                                    ;DCT11 ENABLE CLOCK,AND THEN SET UP CLOCK FOR 80 MS PERIODE
3034                                    ;
3035                                    ;
3036                                    ;QBUS WAIT FOR AT LEAST 80 MS AND CHECK IF AN INTERUPT OCCUR
3037                                    ;ON DCT11 SHIP AT VECTOR 130
3038                                    ;
3039                                    ;TURN OF CLOCK, WAIT AGAIN FOR MORE THAN 80 MS AND CHECK THAT NO
3040                                    ;INTERUPT OCCUR
3041                                    ;
3042                                    ;ERROR REPORTING:       BSEL0=200       IF TIMEOUT DURING TEST
3043                                    ;                       BSEL0=100       IF ERROR DURING TEST
3044                                    ;                       BSEL0=TEST NUB  IF NO KMV11 ANSWER
3045                                    ;                       BSEL0=0         IF TEST IS OK
3046                                    ;
3047                                    ;IF ERROR               SEL6=1          IF NO INTERUPT OCCUR
3048                                    ;                       SEL6=2          IF BAD VECTOR
3049                                    ;                       SEL6=4          IF INTERUPT OCCUR WHEN CLOCK
3050                                    ;                                       IS NOT ENABLE
3051                                    ;                       SEL6=10         INTERUPT OCCUR TOO EARLY
3052                                    ;
3053                                    ;
3054                                    ;
3055                                    ;                       SEL2=EXPECTED VECTOR
3056                                    ;                       SEL4=OBTAINED VECTOR
3057                                    ;
3058                                    ;
3059                                    ; MICRO TEST NB= 27
3060
3061 024166                             STARS 1
3062
3063
3064
3065
3066
3067
3068
3069 024166                             BGNTST
```

```
3070 024166  004737  014400                    JSR     PC,CLRKMV       ;CLR REG
3071 024172  004737  014502                    JSR     PC,MAINM1       ;SET MAINT MODE
3072 024176  004537  014552          RTCLK:    JSR     R5,TSTNUB
3073 024202  000027                            .WORD   27
3074
3075 024204                                    WAITB   0,2             ;WAIT FOR TEST EXECUTION
3076
3077
3078 024224  004737  013102                    JSR     PC,TSTERR       ;CHECK BSEL0
3079 024230  000520                            BR      1$              ;TEST OK
3080 024232  000423                            BR      2$              ;TIMEOUT ERROR
3081 024234  000432                            BR      3$              ;NO KMV ANSWER
3082
3083
3084 024236  022777  000001  166216            CMP     #1,@KMVP06      ;ERROR DURING TEST ,SEE WHICH ONE
3085 024244  001436                            BEQ     4$              ;NO INTERUPT OCCUR
3086
3087 024246  022777  000002  166206            CMP     #2,@KMVP06
3088 024254  001442                            BEQ     5$              ;INT ON BAD VECTOR
3089
3090 024256  022737  000004  012462            CMP     #4,KMVP06
3091 024264  001452                            BEQ     6$              ;INT OCCUR WHEN CLOCK IS DESABLE
3092
3093
3094
3095 024266  022737  000010  012462            CMP     #10,KMVP06      ;INTERUPT OCCUR TOO EARLY
3096 024274  001456                            BEQ     7$
3097
3098 024276  000137  024452                    JMP     10$             ;WRONG KMV11 ANSWER
3099
3100
3101
3102
3103 024302                          2$:       ERRHRD  8,EM0006        ;TIMEOUT ERROR
3104 024312  004737  012740                    JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
3105 024316                                    ESCAPE  TST
3106
3107
3108
3109 024322                          3$:       ERRHRD  9,EM0004        ;NO KMV11 ANSWER
3110 024332  004737  012740                    JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
3111 024336                                    ESCAPE  TST
3112
3113
3114 024342                          4$:       ERRHRD  10,EM0011       ;NO INTERUPT OCCUR
3115 024352  004737  012740                    JSR     PC,CHKMAX       ;CHECK IF TOO MANY ERROR
3116 024356                                    ESCAPE  TST
3117
3118
3119
3120 024362  017737  166072  012402  5$:       MOV     @KMVP04,VECT    ;READ BAD VECT
3121 024370  012737  000130  002272            MOV     #130,GOOD
3122 024376                                    ERRHRD  11,EM0007,PINTR ;INTERUPT OCCUR AT A BAD VECTOR
3123 024406                                    ESCAPE  TST
3124
3125
3126 024412                          6$:       ERRHRD  12,EM0012       ;INT OCCUR WHEN CHIP IS DESABLE
```

```
3127 024422  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3128 024426                                  ESCAPE  TST
3129
3130
3131
3132
3133 024432                          7$:     ERRHRD  13,EM0023       ;INTERUPT OCCUR TOO EARLY
3134 024442  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3135 024446                                  ESCAPE  TST
3136
3137
3138
3139
3140
3141
3142 024452                          10$:    ERRHRD  14,EM0024       ;UNKNWON ANSWER
3143 024462  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3144 024466                                  ESCAPE  TST
3145
3146
3147
3148
3149 024472  000240                          1$:     NOP
3150 024474                                  ENDTST
3151
3152
3153
```

```
3155 024476                     BADHEAD
                                ;********************** TEST4 **************
3156                            ;BAUD RATE GENERATOR TEST
3157 024476                     BADHEAD
                                ;********************** TEST4 **************
3158
3159
3160
3161
3162
3163
3164
3165
3166 024476                     STARS 1
3167                            ;THIS TEST READ THE STATUS AND THE OUTPUT OF THE BAUD RATE GENERATOR
3168                            ;DURING EACH PHASE OF THE CLOCK PULSE
3169                            ;
3170                            ;
3171                            ;NOTE:THIS TEST AND ALL ITS VERIFICATIONS ARE MADE BY THE DCT11 WHICH
3172                            ;       ONLY GIVE TEST RESULT TO THE HOST VIA CSR'S
3173                            ;       ALL THE TIMING IS CHECKED BY THE DCT11
3174                            ;
3175                            ;
3176                            ;
3177                            ;TEST DESCRIPTION:
3178                            ;       -DCT11 LOAD GENE COUNT WITH MAX COUNT (=^4.74 MSEC)
3179                            ;
3180                            ;       -READ BACK GENE COUNT AND STATUS AND CHECK
3181                            ;
3182                            ;
3183
3184
3185                            STEP 1:READ COUNT AFTER STARTING CLOCK
3186                                   CLOCK COUNT MUST BE NEGATIVE
3187                                   OUTPUT MUST BE = 1
3188
3189
3190
3191                                    ERROR REPORTING:
3192                                    IF COUNT=POSITIVE       BSEL0=100=ERROR
3193                                                            SEL6 =1  =GENE COUNT CAN'T BE READ OR
3194                                                                      WRITE CORRECTLY
3195                                    IF OUTPUT=0             BSEL0=100=ERROR
3196                                                            SEL6 =2  =GENE OUTPUT ISN'T IN A GOOD
3197                                                                      STATE(NO ACTION)
3198
3199                            STEP 2:  WAIT 2.5MSEC AND READ BACK AGAIN GENERATOR COUNT AND STATUS
3200                                     OUTPUT MUST BE = 0
3201
3202
3203
3204
3205                                    ERROR REPORTING:
3206                                    IF OUTPUT =1            BSEL0=100= ERROR
3207                                                            SEL6=10  =OUTPUT ISN'T IN A GOOD STATE
3208
3209                            STEP3:WAIT 2.5 MSEC MORE AND READ BACK AGAIN GENERATOR COUNT AND STATUS
```

```
                                ;               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
```

```
3228 024476                                BGNTST
3229 024476                                BGNSUB
3230 024500  004737  014400                       JSR     PC,CLRKMV       ;CLR REG
3231 024504  004737  014502                       JSR     PC,MAINM1       ;SET MAINT MODE
3232 024510  004537  014552                       JSR     R5,TSTNUB
3233 024514  000030                              .WORD   30
3234
3235
3236 024516                                BDRGEN: WAITB   0,1             ;WAIT FOR TEST EXECUTION
3237
3238 024536  004737  013102                       JSR     PC,TSTERR       ;CHECK BSEL0 TO SEE IF ERROR
3239 024542  000137  024742                       JMP     BDROK0          ;TEST OK BR AT END
3240 024546  000402                              BR      2$              ;TIME OUT ERROR
3241 024550  000401                              BR      2$              ;NO KMV11 ANSWER
3242 024552  000410                              BR      3$              ;ERROR DURING TEST
3243
3244
3245
3246 024554                                2$:     ERRHRD  15,EM0004       ;NO KMV11 ANSWER
3247 024564  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3248 024570                                       ESCAPE SUB
3249
3250
3251
3252 024574                                3$:                             ;LOOK WHICH ERROR
3253 024574  017737  165662  002326               MOV     @KMVP06,SEL6    ;READ SEL6
3254 024602  022737  000001  002326               CMP     #1,SEL6         ;LOOK IF ERROR 1
3255 024610  001010                              BNE     4$              ;NO
3256
3257 024612                                       ERRHRD  16,EM0012       ;GENE COUNT CAN'T BE READ OR WRITTE CORRECTLY
3258 024622  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3259 024626                                       ESCAPE SUB
3260
3261
3262 024632  022737  000002  002326  4$:         CMP     #2,SEL6         ;LOOK IF ERROR 2
3263 024640  001010                              BNE     5$              ;NO
3264
3265
3266
3267 024642                                       ERRHRD  17,EM0013       ;GENE OUTPUT ISN'T IN A GOOD STATE
3268 024652  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3269 024656                                       ESCAPE SUB
3270
3271
3272
3273 024662  022737  000010  002326  5$:         CMP     #10,SEL6        ;EROR10?
3274 024670  001414                              BEQ     GENOUT
3275 024672  022737  000040  002326               CMP     #40,SEL6
3276 024700  001410                              BEQ     GENOUT
3277
3278 024702                                       ERRHRD  18,EM0024       ;WRONG KMV11 ANSWER
3279 024712  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3280 024716                                       ESCAPE  SUB
3281
3282
3283
3284
```

```
3285
3286
3287 024722                          GENOUT: ERRHRD  19,EM0014       ;NO ACTION ON GENERATOR OUTPUT
3288 024732  004737  012740                  JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3289 024736                                  ESCAPE SUB
3290
3291
3292
3293 024742                          BDROK0:
3294 024742                          ENDSUB
```

```
3296
3297 024744                                  BGNSUB
3298 024746  004737  014400                          JSR     PC,CLRKMV       ;CLR REG
3299 024752  004737  014502                          JSR     PC,MAINM1       ;SET MAINT MODE
3300 024756  004537  014552                          JSR     R5,TSTNUB
3301 024762  000031                                  .WORD   31
3302
3303
3304 024764                                          WAITB   0,1             ;WAIT FOR TEST EXECUTION
3305
3306 025004  004737  013102                          JSR     PC,TSTERR       ;CHECK BSEL0 TO SEE IF ERROR
3307 025010  000137  025210                          JMP     BDROK1          ;TEST OK BR AT END
3308 025014  000402                                  BR      2$              ;TIME OUT ERROR
3309 025016  000401                                  BR      2$              ;NO KMV11 ANSWER
3310 025020  000410                                  BR      3$              ;ERROR DURING TEST
3311
3312
3313
3314 025022                                  2$:     ERRHRD  20,EM0004       ;NO KMV11 ANSWER
3315 025032  004737  012740                          JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3316 025036                                          ESCAPE SUB
3317
3318
3319
3320 025042                                  3$:                             ;LOOK WHICH ERROR
3321 025042  017737  165414  002326                  MOV     @KMVP06,SEL6    ;READ SEL6
3322 025050  022737  000001  002326                  CMP     #1,SEL6         ;LOOK IF ERROR 1
3323 025056  001010                                  BNE     4$              ;NO
3324
3325 025060                                          ERRHRD  21,EM0112       ;GENE COUNT CAN'T BE READ OR WRITTE CORRECTLY
3326 025070  004737  012740                          JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3327 025074                                          ESCAPE SUB
3328
3329
3330 025100  022737  000002  002326  4$:     CMP     #2,SEL6         ;LOOK IF ERROR 2
3331 025106  001010                                  BNE     5$              ;NO
3332
3333
3334
3335 025110                                          ERRHRD  22,EM0113       ;GENE OUTPUT ISN'T IN A GOOD STATE
3336 025120  004737  012740                          JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3337 025124                                          ESCAPE SUB
3338
3339
3340
3341 025130  022737  000010  002326  5$:     CMP     #10,SEL6        ;EROR10?
3342 025136  001414                                  BEQ     GEN0
3343 025140  022737  000040  002326                  CMP     #40,SEL6
3344 025146  001410                                  BEQ     GEN0
3345
3346 025150                                          ERRHRD  23,EM0024       ;WRONG KMV11 ANSWER
3347 025160  004737  012740                          JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3348 025164                                          ESCAPE  SUB
3349
3350
3351
3352
```

```
3353
3354
3355 025170                            GEN0:   ERRHRD  24,EM0014       ;NO ACTION ON GENERATOR OUTPUT
3356 025200  004737  012740                    JSR     PC,CHKMAX              ;CHECK IF TOO MANY ERROR
3357 025204                                    ESCAPE SUB
3358
3359
3360
3361 025210                            BDROK1:
3362 025210                            ENDSUB
3363 025212                            ENDTST
```

```
3365 025214                         BADHEAD
                                    ;********************** TEST5 **************
3366                                ;TRANSMIT DIFFERENT FRAMES (OF 500 WORDS) AT 2,4 KBAUDS SPEED IN INTERNAL
3367                                ;MODE WHITHOUT ANY INTERUPT ON CHANNEL A
3368 025214                         BADHEAD
                                    ;********************** TEST5 **************
3369
3370
3371
3372
3373
3374
3375
3376 025214                         STARS 1
3377                                ;QBUS WRITTE DIFFERENT TX TABLE OF 500 WORDS, LOAD IN KMV11 CSR'S
3378                                ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
3379                                ;
3380                                ;
3381                                ;
3382                                ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA  AND WRITTE BACK
3383                                ;IN RX TABLE (TRANSFER FROM QBUS TO KMV11 =DMA)
3384                                ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
3385                                ;RX TABLE =TX TABLE
3386                                ;
3387                                ;
3388                                ;PARAMETERS SELECTION:
3389                                ;       SEL2=   TX TABLE ADDRESS
3390                                ;       SEL4=   TX TABLE LENGTH
3391                                ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
3392                                ;       BSEL7=      "       "     RX   "
3393                                ;       SEL12=  RX TABLE ADDRESS
3394                                ;       SEL14=  SPEED SELECTION
3395                                ;       SEL16=  ERROR STATUS
3396                                ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
3397                                ;                                       >0 IF TX>RX,<0 IF TX<RX
3398                                ;       BSEL0=  TEST STATUS
3399                                ;
3400                                ;
3401                                ;
3402                                ;TEST STATUS DESCRIPTION:
3403                                ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
3404                                ;       BSEL0=  200     =TIMEOUT ERROR
3405                                ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
3406                                ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
3407                                ;
3408                                ;
3409                                ;
3410                                ;ERROR STATUS DESCRIPTION:
3411                                ;
3412                                ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
3413                                ;
3414                                ;
3415                                ;       SEL16=  BIT14=1 =FCS ERROR
3416                                ;       SEL16=  BIT13=1 =OVERRUN ERROR
3417                                ;       SEL16=  BIT8 =1 =ILLEGAL INTERUPT ERROR
3418                                ;       SEL16=  BIT7 =1 =RX ABORT ERROR
3419                                ;       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 36        =TRANSMIT  FRAMES AT 2,4KB SPEED ON CHANNEL A WHITHOUT INTERUPT
                                ;
                                ;
 025214                         STARS 1
```

```
3433 025214                              BGNTST
3434 025214  004737  014400                      JSR     PC,CLRKMV               ;CLR REG
3435 025220  005037  002250                      CLR     FLAG
3436 025224  004737  014502                      JSR     PC,MAINM1               ;SET MAINT MODE
3437 025230  012737  000500  012416              MOV     #500,LENGTH             ;SELECT LENGTH
3438
3439 025236  012737  005473  012414              MOV     #KB2.4,TSPEED           ;SELECT SPEED
3440
3441 025244  012703  000001          INTTX:  MOV     #1,R3                   ;SELECT A PATTERN
3442
3443
3444 025250  005203                  TXSTAR: INC     R3                      ;NEW ONE
3445 025252  013704  012416                      MOV     LENGTH,R4               ;LOAD LENGTH
3446 025256  012702  002370                      MOV     #TTABLE,R2              ;TX TABLE ADDRESS
3447 025262  004737  013160          10$:    JSR     PC,GENER                ;WRTTE TX TABLE
3448 025266  013722  012400                      MOV     DATA,(R2)+
3449 025272  005304                              DEC     R4                      ;ALL DONE?
3450 025274  001372                              BNE     10$
3451
3452
3453
3454 025276  013704  012416                      MOV     LENGTH,R4
3455 025302  012702  006370                      MOV     #RTABLE,R2              ;CLEAR RX TABLE
3456 025306  005022                  11$:    CLR     (R2)+
3457 025310  005304                              DEC     R4
3458 025312  001375                              BNE     11$
3459
3460
3461
3462
3463 025314  013777  012414  165146              MOV     TSPEED,@KMVP14          ;SEND TX SPEED
3464 025322  012777  002370  165126              MOV     #TTABLE,@KMVP02         ;SEND TX TABLE ADDRESS
3465 025330  013777  012416  165122              MOV     LENGTH,@KMVP04          ;LOAD TX TABLE ADDRESS
3466 025336  012777  006370  165122              MOV     #RTABLE,@KMVP12         ;LOAD RX TABLE ADDRESS
3467 025344  005077  165112                      CLR     @KMVP06
3468
3469
3470
3471
3472
3473
3474 025350  004537  014552                      JSR     R5,TSTNUB
3475 025354  000036                              .WORD   36                      ;DO TEST 36= CHA TEST
3476
3477
3478
3479 025356                          2$:     WAITB   0,13                    ;WAIT FOR TEST EXECUTION
3480
3481
3482 025376  004737  013102                      JSR     PC,TSTERR               ;CHECK BSEL0
3483
3484 025402  000427                              BR      6$                      ;TEST OK CHECK RX TABLE
3485 025404  000402                              BR      3$                      ;TIMEOUT ERROR
3486 025406  000401                              BR      3$                      ;NO KMV11 ANSWER
3487 025410  000410                              BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
3488
3489
```

```
3490 025412                          3$:    ERRHRD  25,EM0004          ;NO KMV11 ANSWER
3491 025422  004737  012740                 JSR     PC,CHKMAX          ;CHECK IF TOO MANY ERROR
3492 025426                                 ESCAPE  TST
3493
3494
3495
3496 025432                          4$:                                ;ERROR DURING TEST READ ERROR STATUS
3497                                                                    ;TO CHECK WHICH ONE
3498
3499 025432  017737  165034  012424        MOV     @KMVP16,STAERR     ;READ ERROR STATUS
3500
3501 025440  017737  165020  012426        MOV     @KMVP10,WRDCNT     ;READ WORD COUNT DISCREPANCY
3502
3503 025446                                 ERRHRD  26,EM0015,PRSTER   ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3504                                                                    ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3505 025456                                 ESCAPE  TST
3506
3507
3508
3509
3510
3511
3512
3513
3514 025462  012702  002370          6$:    MOV     #TTABLE,R2         ;LOAD TXTABLE ADDRESS
3515 025466  012705  006370                 MOV     #RTABLE,R5         ;   "  RXTABLE ADDRESS
3516 025472  013704  012416                 MOV     LENGTH,R4          ;TABLE LENGTH
3517
3518 025476  022225                  RXCK:  CMP     (R2)+,(R5)+        ;CHECK RX AND TX TABLE
3519 025500  001007                         BNE     RXERR
3520 025502  005304                         DEC     R4                 ;ALL CHECK?
3521 025504  001374                         BNE     RXCK               ;NO BRANCH
3522
3523
3524
3525 025506  022703  000005                CMP     #5,R3              ;ALL KIND OF PATTERN DONE?
3526 025512  001256                         BNE     TXSTAR             ;NO TRY WHITH NEW ONE
3527
3528
3529
3530 025514  000137  025636                JMP     RXEND
3531
3532 025520  162705  000002          RXERR: SUB     #2,R5
3533 025524  162702  000002                 SUB     #2,R2
3534
3535 025530  011237  012410                MOV     (R2),TXDATA
3536 025534  011537  012412                MOV     (R5),RXDATA
3537
3538 025540  005737  002250                TST     FLAG               ;LOOK IF 1ST ERROR
3539 025544  001014                         BNE     7$
3540
3541 025546                                 ERRHRD  27,EM0015,PFRAME   ;DATA CMP ERROR
3542 025556  005237  002250                INC     FLAG
3543 025562  062702  000002                ADD     #2,R2              ;POINT NEXT ADDRESS
3544 025566  062705  000002                ADD     #2,R5
3545 025572  000137  025476                JMP     RXCK
3546
```

```
3547 025576                          7$:     ERRHRD  27,0,PRAMEF            ;SHORT REPORT
3548 025606  005237  002250                  INC     FLAG
3549 025612  062702  000002                  ADD     #2,R2
3550 025616  062705  000002                  ADD     #2,R5                   ;POINT NEXT ADDRESS
3551 025622  022737  000010  002250          CMP     #10,FLAG                ;LOOK IF 10 REPORT
3552 025630  001322                          BNE     RXCK
3553
3554 025632                                  ESCAPE  TST
3555
3556
3557 025636                          RXEND:
3558
3559
3560
3561 025636                          ENDTST
```

```
3563 025640                         BADHEAD
                                    ;********************** TEST6 **************
3564                                ;TRANSMIT DIFFERENT FRAMES (OF 500 WORDS) AT 2,4 KBAUDS SPEED IN INTERNAL
3565                                ;MODE WHITHOUT ANY INTERUPT ON CHANNEL B
3566 025640                         BADHEAD
                                    ;********************** TEST6 **************
3567
3568
3569
3570
3571
3572
3573
3574 025640                         STARS 1
3575                                ;QBUS WRITTE DIFFERENT TX TABLE OF 500 WORDS, LOAD IN KMV11 CSR'S
3576                                ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
3577                                ;
3578                                ;
3579                                ;
3580                                ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHB AND WRITTE BACK
3581                                ;IN RX TABLE (TRANSFER FROM QBUS TO KMV11 =DMA)
3582                                ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
3583                                ;RX TABLE =TX TABLE
3584                                ;
3585                                ;
3586                                ;PARAMETERS SELECTION:
3587                                ;       SEL2=   TX TABLE ADDRESS
3588                                ;       SEL4=   TX TABLE LENGTH
3589                                ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
3590                                ;       BSEL7=      "       "       RX   "
3591                                ;       SEL12=  RX TABLE ADDRESS
3592                                ;       SEL14=  SPEED SELECTION
3593                                ;       SEL16=  ERROR STATUS
3594                                ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
3595                                ;                                       >0 IF TX>RX,<0 IF TX<RX
3596                                ;       BSEL0=  TEST STATUS
3597                                ;
3598                                ;
3599                                ;
3600                                ;TEST STATUS DESCRIPTION:
3601                                ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
3602                                ;       BSEL0=  200     =TIMEOUT ERROR
3603                                ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
3604                                ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
3605                                ;
3606                                ;
3607                                ;
3608                                ;ERROR STATUS DESCRIPTION:
3609                                ;
3610                                ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
3611                                ;
3612                                ;
3613                                ;       SEL16=  BIT14=1 =FCS ERROR
3614                                ;       SEL16=  BIT13=1 =OVERRUN ERROR
3615                                ;       SEL16=  BIT8 =1 =ILLEGAL INTERUPT ERROR
3616                                ;       SEL16=  BIT7 =1 =RX ABORT ERROR
3617                                ;       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 AT 2,4KB SPEED ON CHANNEL B WHITHOUT INTERUPT
                                ;
                                ;
3629 025640                     STARS 1
```

```
3631 025640                              BGNTST
3632 025640  004737  014400                      JSR     PC,CLRKMV               ;CLR REG
3633 025644  005037  002250                      CLR     FLAG
3634 025650  004737  014502                      JSR     PC,MAINM1               ;SET MAINT MODE
3635 025654  012737  000500  012416              MOV     #500,LENGTH             ;SELECT LENGTH
3636
3637 025662  012737  005473  012414              MOV     #KB2.4,TSPEED           ;SELECT SPEED
3638
3639 025670  012703  000001                      MOV     #1,R3                   ;SELECT A PATTERN
3640
3641
3642 025674  005203                    BTXSTA:   INC     R3                      ;NEW ONE
3643 025676  013704  012416                      MOV     LENGTH,R4               ;LOAD LENGTH
3644 025702  012702  002370                      MOV     #TTABLE,R2              ;TX TABLE ADDRESS
3645 025706  004737  013160            10$:      JSR     PC,GENER                ;WRTTE TX TABLE
3646 025712  013722  012400                      MOV     DATA,(R2)+
3647 025716  005304                              DEC     R4                      ;ALL DONE?
3648 025720  001372                              BNE     10$
3649
3650
3651
3652 025722  013704  012416                      MOV     LENGTH,R4
3653 025726  012702  006370                      MOV     #RTABLE,R2              ;CLEAR RX TABLE
3654 025732  005022                    11$:      CLR     (R2)+
3655 025734  005304                              DEC     R4
3656 025736  001375                              BNE     11$
3657
3658
3659
3660
3661 025740  013777  012414  164522              MOV     TSPEED,@KMVP14          ;SEND TX SPEED
3662 025746  012777  002370  164502              MOV     #TTABLE,@KMVP02         ;SEND TX TABLE ADDRESS
3663 025754  013777  012416  164476              MOV     LENGTH,@KMVP04          ;LOAD TX TABLE ADDRESS
3664 025762  012777  006370  164476              MOV     #RTABLE,@KMVP12         ;LOAD RX TABLE ADDRESS
3665 025770  005077  164466                      CLR     @KMVP06
3666
3667
3668
3669
3670
3671 025774  004537  014552            1$:       JSR     R5,TSTNUB
3672 026000  000037                              .WORD   37                      ;DO TEST 37= CHB TEST
3673
3674
3675
3676 026002                            2$:       WAITB   0,13                    ;WAIT FOR TEST EXECUTION
3677
3678
3679 026022  004737  013102                      JSR     PC,TSTERR               ;CHECK BSEL0
3680
3681 026026  000427                              BR      6$                      ;TEST OK CHECK RX TABLE
3682 026030  000402                              BR      3$                      ;TIMEOUT ERROR
3683 026032  000401                              BR      3$                      ;NO KMV11 ANSWER
3684 026034  000410                              BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
3685
3686
3687 026036                            3$:       ERRHRD  25,EM0004               ;NO KMV11 ANSWER
```

```
3688 026046  004737  012740                JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3689 026052                                ESCAPE  TST
3690
3691
3692
3693 026056                          4$:                                   ;ERROR DURING TEST READ ERROR STATUS
3694                                                                       ;TO CHECK WHICH ONE
3695
3696 026056  017737  164410  012424        MOV     @KMVP16,STAERR          ;READ ERROR STATUS
3697
3698 026064  017737  164374  012426        MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
3699
3700 026072                                ERRHRD  26,EM0115,PRSTER        ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3701                                                                       ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3702 026102                                ESCAPE  TST
3703
3704
3705
3706
3707
3708 026106  012702  002370          6$:   MOV     #TTABLE,R2              ;LOAD TXTABLE ADDRESS
3709 026112  012705  006370                MOV     #RTABLE,R5              ;   ''  RXTABLE ADDRESS
3710 026116  013704  012416                MOV     LENGTH,R4               ;TABLE LENGTH
3711
3712 026122  022225                BRXCK:  CMP     (R2)+,(R5)+             ;CHECK RX AND TX TABLE
3713 026124  001007                        BNE     BRXERR
3714 026126  005304                        DEC     R4                      ;ALL CHECK?
3715 026130  001374                        BNE     BRXCK                   ;NO BRANCH
3716
3717
3718
3719 026132  022703  000005                CMP     #5,R3                   ;ALL KIND OF PATTERN DONE?
3720 026136  001256                        BNE     BTXSTA                  ;NO TRY WHITH NEW ONE
3721
3722
3723
3724 026140  000137  026262                JMP     BRXEND
3725
3726 026144  162705  000002          BRXERR: SUB     #2,R5
3727 026150  162702  000002                SUB     #2,R2
3728
3729 026154  011237  012410                MOV     (R2),TXDATA
3730 026160  011537  012412                MOV     (R5),RXDATA
3731
3732 026164  005737  002250                TST     FLAG                    ;LOOK IF 1ST ERROR
3733 026170  001014                        BNE     7$
3734
3735 026172                                ERRHRD  27,EM0115,PFRAME        ;DATA CMP ERROR
3736 026202  005237  002250                INC     FLAG
3737 026206  062702  000002                ADD     #2,R2                   ;POINT NEXT ADDRESS
3738 026212  062705  000002                ADD     #2,R5
3739 026216  000137  025476                JMP     RXCK
3740
3741 026222                          7$:   ERRHRD  27,0,PRAMEF             ;SHORT REPORT
3742 026232  005237  002250                INC     FLAG
3743 026236  062702  000002                ADD     #2,R2
3744 026242  062705  000002                ADD     #2,R5                   ;POINT NEXT ADDRESS
```

```
3745 026246  022737  000010  002250          CMP     #10,FLAG                ;LOOK IF 10 REPORT
3746 026254  001322                          BNE     BRXCK
3747
3748 026256                                  ESCAPE  TST
3749
3750
3751 026262                          BRXEND:
3752
3753
3754
3755 026262                          ENDTST
```

```
3757
3758
3759
3760 026264                             BADHEAD
                                        ;********************** TEST7 **************
3761                                    ;TRANSMIT DIFFERENT  FRAME OF VARIOUS LENGTH (FROM 2BYTES TO 2K BYTES)
3762                                    ;AT 56KBAUDS IN INTERNAL  MODE ON CHANNEL A  (TRANSMISSION WITH INTERUPT)
3763 026264                             BADHEAD
                                        ;********************** TEST7 **************
3764
3765
3766
3767
3768
3769
3770
3771 026264                             STARS 1
3772                                    ;QBUS WRITTE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
3773                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED
3774                                    ;
3775                                    ;
3776                                    ;
3777                                    ;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA AND WRITTE BACK
3778                                    ;IN RX TABLE
3779                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
3780                                    ;RX TABLE =TX TABLE
3781                                    ;SPEED=56KBAUDS
3782                                    ;
3783                                    ;
3784                                    ;PARAMETERS SELECTION:
3785                                    ;       SEL2=   TX TABLE ADDRESS
3786                                    ;       SEL4=   TX TABLE LENGTH
3787                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
3788                                    ;       BSEL7=     "          "     RX   "
3789                                    ;       SEL12=  RX TABLE ADDRESS
3790                                    ;       SEL14=  SPEED SELECTION
3791                                    ;       SEL16=  ERROR STATUS
3792                                    ;       BSEL0=  TEST STATUS
3793                                    ;
3794                                    ;
3795                                    ;
3796                                    ;TEST STATUS DESCRIPTION:
3797                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
3798                                    ;       BSEL0=  200     =TIMEOUT ERROR
3799                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
3800                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
3801                                    ;
3802                                    ;
3803                                    ;
3804                                    ;ERROR STATUS DESCRIPTION:
3805                                    ;
3806                                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERRGR STATUS AND WORD COUNT DISCREPANCY
3807                                    ;
3808                                    ;
3809                                    ;       SEL16=  BIT14=1 =FCS ERROR
3810                                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
3811                                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERUPT 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  ABLE (USE
                                ;                                                ONLY DURING SEL  TEST)
                                ;
                                ;
                                ;MICRO DIAG TEST DESCRIPTION:
                                ;TEST 40        =TRANSMIT VARIOUS LENGTH FRAME  AT 56 KBAUDS ON CHANNEL A
                                ;
                                ;
026264                          STARS 1
```

```
3828 026264                                BGNTST
3829 026264  004737  014400                       JSR     PC,CLRKMV          ;CLR REG
3830 026270  004737  014502                       JSR     PC,MAINM1          ;SET MAINT MODE
3831 026274  005037  002250                       CLR     FLAG
3832
3833
3834 026300  012703  000005                       MOV     #5,R3              ;SELECT RANDOM PATTERN
3835 026304  012737  000174  012414               MOV     #KB56,TSPEED       ;SELECT SPEED
3836
3837 026312  012737  000001  012416  TXLIAR:  MOV     #1,LENGTH          ;START WITH 2 CHARACTERS
3838
3839 026320  013704  012416           TXLBGN:  MOV     LENGTH,R4
3840 026324  012702  002370                    MOV     #TTABLE,R2
3841 026330                                    BREAK
3842 026332  004737  013160           10$:     JSR     PC,GENER           ;WRITE TTABLE
3843 026336  013722  012400                    MOV     DATA,(R2)+
3844 026342  005304                           DEC     R4
3845 026344  001372                           BNE     10$
3846
3847
3848 026346  013704  012416                    MOV     LENGTH,R4          ;CLEAR RX TABLE
3849 026352  012702  006370                    MOV     #RTABLE,R2
3850 026356  005022           20$:     CLR     (R2)+
3851 026360  005304                           DEC     R4
3852 026362  001375                           BNE     20$
3853
3854
3855
3856
3857
3858
3859 026364  013777  012414  164076           MOV     TSPEED,@KMVP14     ;SEND TX SPEED
3860 026372  012777  002370  164056           MOV     #TTABLE,@KMVP02    ;   "  TX TABLE ADDRESS
3861 026400  013777  012416  164052           MOV     LENGTH,@KMVP04     ;   "  "     "  LENGTH
3862 026406  012777  006370  164052           MOV     #RTABLE,@KMVP12    ;SEND RX TABLE ADDRESS
3863 026414  005077  164042                    CLR     @KMVP06            ;CLR EXTENDED ADDRESS
3864
3865
3866
3867
3868 026420  004537  014552                    JSR     R5,TSTNUB
3869 026424  000040                           .WORD   40                 ;DO TEST 40= CHA TEST
3870
3871
3872
3873 026426                           2$:      WAITB   0,2                ;WAIT FOR TEST EXECUTION
3874
3875
3876 026446  004737  013102                    JSR     PC,TSTERR          ;CHECK BSEL0
3877
3878 026452  000427                           BR      6$                 ;TEST OK CHECK RX TABLE
3879 026454  000402                           BR      3$                 ;TIMEOUT ERROR
3880 026456  000401                           BR      3$                 ;NO KMV11 ANSWER
3881 026460  000410                           BR      4$                 ;CHECK SEL16 TO SEE WHICH ONE
3882
3883
3884 026462                           3$:      ERRHRD  28,EM0004          ;NO KMV11 ANSWER
```

```
3885 026472  004737  012740                JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
3886 026476                                ESCAPE  TST
3887
3888 026502                          4$:                                   ;ERROR DURING TEST READ ERROR STATUS
3889                                                                       ;TO CHECK WHICH ONE
3890
3891 026502  017737  163764  012424        MOV     @KMVP16,STAERR          ;READ ERROR STATUS
3892
3893 026510  017737  163750  012426        MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
3894
3895 026516                                ERRHRD  29,EM0016,PRSTER        ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3896                                                                       ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3897 026526                                ESCAPE  TST
3898
3899
3900
3901
3902 026532  012702  002370          6$:   MOV     #TTABLE,R2              ;LOAD TX TABLE ADDRESS
3903 026536  012705  006370                MOV     #RTABLE,R5              ; ''   RX    ''      ''
3904 026542  013704  012416                MOV     LENGTH,R4               ; ''   TX TABLE LENGTH
3905
3906
3907 026546  022522                RXLCK:  CMP     (R5)+,(R2)+             ;CMP TX AND RX TABLE
3908 026550  001015                        BNE     RXLERR                  ;BR IF ERROR
3909 026552  005304                        DEC     R4                      ;ALL DONE
3910 026554  001374                        BNE     RXLCK                   ;NO
3911
3912 026556  062737  000400  012416        ADD     #400,LENGTH             ;CHANGE LENGTH
3913 026564  022737  002000  012416        CMP     #2000,LENGTH            ;IS IT MAX?
3914 026572  100252                        BPL     TXLBGN                  ;NO DO TEST AGAIN WHITH NEW TABLE
3915                                                                       ;                          LENGTH
3916
3917 026574  005303                        DEC     R3                      ;SELECT OTHER PATERNS
3918 026576  001245                        BNE     TXLTAR
3919
3920 026600  000137  026722                JMP     RXLEND
3921
3922
3923
3924 026604  162705  000002          RXLERR: SUB     #2,R5
3925 026610  162702  000002                SUB     #2,R2
3926
3927 026614  011237  012410                MOV     (R2),TXDATA
3928 026620  011537  012412                MOV     (R5),RXDATA
3929
3930 026624  005737  002250                TST     FLAG                    ;LOOK IF 1ST ERROR
3931 026630  001014                        BNE     30$
3932
3933 026632                                ERRHRD  30,EM0016,PFRAME        ;DATA CMP ERROR
3934 026642  005237  002250                INC     FLAG
3935 026646  062702  000002                ADD     #2,R2                   ;POINT NEXT ADDRESS
3936 026652  062705  000002                ADD     #2,R5
3937 026656  000137  025476                JMP     RXCK
3938
3939 026662                          30$:  ERRHRD  30,0,PRAMEF             ;SHORT REPORT
3940 026672  005237  002250                INC     FLAG
3941 026676  062702  000002                ADD     #2,R2
```

```
3942 026702  062705  000002                ADD     #2,R5                ;POINT NEXT ADDRESS
3943 026706  022737  000010  002250        CMP     #10,FLAG             ;LOOK IF 10 REPORT
3944 026714  001314                        BNE     RXLCK
3945
3946 026716                                ESCAPE  TST
3947
3948
3949
3950
3951 026722                          RXLEND:
3952 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 INTERUPT)
                                BADHEAD
                                ;********************** TEST8 **************

                                STARS 1
                                ;QBUS WRITTE 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 WRITTE 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 INTERUPT 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 41        =TRANSMIT VARIOUS LENGTH FRAME  AT 56 KBAUDS ON CHANNEL B
                                ;
                                ;
                                STARS 1
```

```
4026 026724                                BGNTST
4027 026724  004737  014400                       JSR     PC,CLRKMV               ;CLR REG
4028 026730  004737  014502                       JSR     PC,MAINM1               ;SET MAINT MODE
4029 026734  005037  002250                       CLR     FLAG
4030
4031
4032 026740  012703  000005                       MOV     #5,R3                   ;SELECT RANDOM PATTERN
4033 026744  012737  000174  012414               MOV     #KB56,TSPEED            ;SELECT SPEED
4034
4035 026752  012737  000001  012416 BXLTAR:      MOV     #1,LENGTH               ;START WITH 2 CHARACTERS
4036
4037 026760  013704  012416        BXLBGN:      MOV     LENGTH,R4
4038 026764  012702  002370                       MOV     #TTABLE,R2
4039 026770                                      BREAK
4040 026772  004737  013160        10$:         JSR     PC,GENER                ;WRITE TTABLE
4041 026776  013722  012400                       MOV     DATA,(R2)+
4042 027002  005304                              DEC     R4
4043 027004  001372                              BNE     10$
4044
4045
4046 027006  013704  012416                       MOV     LENGTH,R4               ;CLEAR RX TABLE
4047 027012  012702  006370                       MOV     #RTABLE,R2
4048 027016  005022        20$:         CLR     (R2)+
4049 027020  005304                              DEC     R4
4050 027022  001375                              BNE     20$
4051
4052
4053
4054
4055
4056
4057 027024  013777  012414  163436               MOV     TSPEED,@KMVP14          ;SEND TX SPEED
4058 027032  012777  002370  163416               MOV     #TTABLE,@KMVP02         ;  "  TX TABLE ADDRESS
4059 027040  013777  012416  163412               MOV     LEN" H,@KMVP04          ;  "  "    "   LENGTH
4060 027046  012777  006370  163412               MOV     #RTABLE,@KMVP12         ;SEND RX TABLE ADDRESS
4061 027054  005077  163402                       CLR     @KMVP06                 ;CLR EXTENDED ADDRESS
4062
4063
4064
4065 027060  004537  014552                       JSR     R5,TSTNUB
4066 027064  000041                              .WORD   41                      ;DO TEST 41= CHB TEST
4067
4068
4069 027066                        2$:          WAITB   0,2                     ;WAIT FOR TEST EXECUTION
4070
4071
4072 027106  004737  013102                       JSR     PC,TSTERR               ;CHECK BSEL0
4073
4074 027112  000427                              BR      6$                      ;TEST OK CHECK RX TABLE
4075 027114  000402                              BR      3$                      ;TIMEOUT ERROR
4076 027116  000401                              BR      3$                      ;NO KMV11 ANSWER
4077 027120  000410                              BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
4078
4079
4080 027122                        3$:          ERRHRD  28,EM0004               ;NO KMV11 ANSWER
4081 027132  004737  012740                       JSR     PC,CHKMAX               ;CHECK IF TOO MANY ERROR
4082 027136                                      ESCAPE  TST
```

```
4083
4084 027142                          4$:                                ;ERROR DURING TEST READ ERROR STATUS
4085                                                                    ;TO CHECK WHICH ONE
4086
4087 027142  017737  163324  012424          MOV     @KMVP16,STAERR     ;READ ERROR STATUS
4088
4089 027150  017737  163310  012426          MOV     @KMVP10,WRDCNT     ;READ WORD COUNT DISCREPANCY
4090
4091 027156                                  ERRHRD  29,EM0116,PRSTER   ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
4092                                                                    ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
4093 027166                                  ESCAPE  TST
4094
4095
4096
4097
4098 027172  012702  002370          6$:     MOV     #TTABLE,R2         ;LOAD TX TABLE ADDRESS
4099 027176  012705  006370                  MOV     #RTABLE,R5         ; ''   RX    ''      ''
4100 027202  013704  012416                  MOV     LENGTH,R4          ; ''   TX TABLE LENGTH
4101
4102
4103 027206  022522                  BXLCK:  CMP     (R5)+,(R2)+        ;CMP TX AND RX TABLE
4104 027210  001015                          BNE     BXLERR             ;BR IF ERROR
4105 027212  005304                          DEC     R4                 ;ALL DONE
4106 027214  001374                          BNE     BXLCK              ;NO
4107
4108 027216  062737  000400  012416          ADD     #400,LENGTH        ;CHANGE LENGTH
4109 027224  022737  002000  012416          CMP     #2000,LENGTH       ;IS IT MAX?
4110 027232  100252                          BPL     BXLBGN             ;NO DO TEST AGAIN WHITH NEW TABLE
4111                                                                    ;                         LENGTH
4112
4113 027234  005303                          DEC     R3                 ;SELECT OTHER PATERNS
4114 027236  001245                          BNE     BXLTAR
4115
4116 027240  000137  027362                  JMP     BXLEND
4117
4118
4119
4120 027244  162705  000002          BXLERR: SUB     #2,R5
4121 027250  162702  000002                  SUB     #2,R2
4122
4123 027254  011237  012410                  MOV     (R2),TXDATA
4124 027260  011537  012412                  MOV     (R5),RXDATA
4125
4126 027264  005737  002250                  TST     FLAG               ;LOOK IF 1ST ERROR
4127 027270  001014                          BNE     30$
4128
4129 027272                                  ERRHRD  30,EM0116,PFRAME   ;DATA CMP ERROR
4130 027302  005237  002250                  INC     FLAG
4131 027306  062702  000002                  ADD     #2,R2              ;POINT NEXT ADDRESS
4132 027312  062705  000002                  ADD     #2,R5
4133 027316  000137  000000G                 JMP     BXCK
4134
4135 027322                          30$:    ERRHRD  30,0,PRAMEF        ;SHORT REPORT
4136 027332  005237  002250                  INC     FLAG
4137 027336  062702  000002                  ADD     #2,R2
4138 027342  062705  000002                  ADD     #2,R5              ;POINT NEXT ADDRESS
4139 027346  022737  000010  002250          CMP     #10,FLAG           ;LOOK IF 10 REPORT
```

```
4140 027354  001314                          BNE     BXLCK
4141
4142 027356                                   ESCAPE  TST
4143
4144
4145
4146
4147 027362                          BXLEND:
4148 027362                          ENDTST
```

```
4150
4151 027364                             BADHEAD
                                        ;********************** TEST9 **************
4152                                    ;TRANSMIT DIFFERENT FRAMES OF VARIOUS LENGTH IN EXTERNAL LOOP BACK
4153                                    ;MODE ON CHANNEL A
4154 027364                             BADHEAD
                                        ;********************** TEST9 **************
4155
4156
4157
4158
4159
4160
4161 027364                             STARS 1
4162                                    ;
4163                                    ;AT BEGINNING OF TEST ,CHECK IF LOOP BACK CONNECTORS ARE INSTALLED
4164                                    ;OR NOT:IF NOT INSTALLED = EXIT TEST AND GIVE ERROR MESSAGE
4165                                    ;                          **********************************
4166                                    ;
4167                                    ;
4168                                    ;
4169                                    ;
4170                                    ;
4171                                    ;QBUS WRITTE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
4172                                    ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED (56KB)
4173                                    ;
4174                                    ;
4175                                    ;
4176                                    ;DCT11 EXECUTE THE TRANSFER IN EXTERNAL MODE ON CHA  AND WRITTE BACK
4177                                    ;IN RX TABLE
4178                                    ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
4179                                    ;RX TABLE =TX TABLE
4180                                    ;
4181                                    ;
4182                                    ;PARAMETERS SELECTION:
4183                                    ;       SEL2=   TX TABLE ADDRESS
4184                                    ;       SEL4=   TX TABLE LENGTH
4185                                    ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
4186                                    ;       BSEL7=      "       "       RX   "
4187                                    ;       SEL12=  RX TABLE ADDRESS
4188                                    ;       SEL14=  SPEED SELECTION
4189                                    ;       SEL16=  ERROR STATUS
4190                                    ;       BSEL0=  TEST STATUS
4191                                    ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
4192                                    ;                                       >0 IF TX>RX,<0 IF TX<RX
4193                                    ;
4194                                    ;
4195                                    ;
4196                                    ;TEST STATUS DESCRIPTION:
4197                                    ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
4198                                    ;       BSEL0=  200     =TIMEOUT ERROR
4199                                    ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
4200                                    ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
4201                                    ;
4202                                    ;
4203                                    ;
4204                                    ;ERROR STATUS DESCR.PTION:
```

```
4205                    ;
4206                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
4207                    ;
4208                    ;
4209                    ;       SEL16=  BIT14=1 =FCS ERROR
4210                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
4211                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERUPT ERROR
4212                    ;       SEL16=  BIT7 =1 =RX ABORT ERROR
4213                    ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
4214                    ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
4215                    ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
4216                    ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
4217                    ;       SEL16= BIT2 =1 =CLOCK PROBLEM
4218                    ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
4219                    ;                                               ONLY DURING SELF TEST)
4220                    ;
4221                    ;
4222                    ;MICRO DIAG TEST DESCRIPTION:
4223                    ;TEST 42        =TRANSMIT VARIOUS LENGTH FRAME AT 56 KBAUDS SPEED ON CHANNEL A
4224                    ;                       IN EXTERNAL LOOP BACK MODE
4225                    ;
4226                    ;
4227                    ;
4228                    ;
4229                    ;CAUTION:
4230                    ;--------
4231                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4232                    ;
4233                    ;
4234                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4235                    ;
4236                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4237                    ;EXTERNAL LOOP BACK CONECTOR
4238                    ;
4239                    ;
4240                    ;
4241                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4242                    ;MODEM CABLE.
4243                    ;
4244                    ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
4245                    ;MODEM CABLE.
4246                    ;
4247 027364            STARS 1
```

```
027364                          BGNTST
027364  004737  014400                  JSR     PC,CLRKMV               ;CLEAR REGISTERS
027370  004737  014600                  JSR     PC,CKKMV                ;LOOK IF LOOP BACK CON INSTALLED?
027374  005737  012474                  TST     LOOP                    ;IS LOOP BIT=1?
027400  001412                          BEQ     BGNTXA                  ;YES GO ON TEST
027402                                  PRINTF  #MLOOP                  ;NO LOOP BACK PLUGGED .THE MODULE
                                                                        ;WILL NOT BE TESTED IS EXTERNAL LOOP
027422  000137  030060                  JMP     RXAEND

027426  004737  014502          BGNTXA: JSR     PC,MAINM1               ;SET MAINT MODE
027432  005037  002250                  CLR     FLAG

027436  012703  000005                  MOV     #5,R3                   ;SELECT RANDOM PATTERN
027442  012737  000174  012414          MOV     #KB56,TSPEED            ;SELECT SPEED

027450  012737  000001  012416  TXATAR: MOV     #1,LENGTH               ;1ST TABLE LENGTH(200 WORDS)

027456  013704  012416          TXABGN: MOV     LENGTH,R4
027462  012702  002370                  MOV     #TTABLE,R2
027466                                  BREAK
027470  004737  013160          10$:    JSR     PC,GENER                ;WRITE TABLE
027474  013722  012400                  MOV     DATA,(R2)+
027500  005304                          DEC     R4
027502  001372                          BNE     10$

027504  013704  012416                  MOV     LENGTH,R4               ;CLEAR RX TABLE
027510  012702  006370                  MOV     #RTABLE,R2
027514  005022                  20$:    CLR     (R2)+
027516  005304                          DEC     R4
027520  001375                          BNE     20$

027522  013777  012414  162740          MOV     TSPEED,@KMVP14          ;SEND TX SPEED
027530  012777  002370  162720          MOV     #TTABLE,@KMVP02         ;  "  TX TABLE ADDRESS
027536  013777  012416  162714          MOV     LENGTH,@KMVP04          ;  "  "    "    LENGTH
027544  012777  006370  162714          MOV     #RTABLE,@KMVP12         ;SEND RX TABLE ADDRESS
027552  005077  162704                  CLR     @KMVP06                 ;CLR EXTENDED ADDRESS

027556  004537  014552          1$:     JSR     R5,TSTNUB
027562  000042                          .WORD   42                      ;DO TEST 42= CHB TEST

027564                          2$:     WAITB   0,3                     ;WAIT FOR TEST EXECUTION
```

```
027604  004737  013102                  JSR     PC,TSTERR               ;CHECK BSELO

027610  000427                          BR      6$                      ;TEST OK CHECK RX TABLE
027612  000402                          BR      3$                      ;TIMEOUT ERROR
027614  000401                          BR      3$                      ;NO KMV11 ANSWER
027616  000410                          BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE

027620                          3$:     ERRHRD  32,EM0004               ;NO KMV11 ANSWER
027630  004737  012740                  JSR     PC,CHKMAX               ;TOO MANY ERROR?
027634                                  ESCAPE  TST

027640                          4$:                                     ;ERROR DURING TEST READ ERROR STATUS
                                                                        ;TO CHECK WHICH ONE

027640  017737  162626  012424          MOV     @KMVP16,STAERR          ;READ ERROR STATUS

027646  017737  162612  012426          MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY

027654                                  ERRHRD  33,EM0017,PRSTER                ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
                                                                        ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
027664                                  ESCAPE  TST

027670  012702  002370          6$:     MOV     #TTABLE,R2              ;LOAD TABLE PARAMETERS
027674  012705  006370                  MOV     #RTABLE,R5
027700  013704  012416                  MOV     LENGTH,R4

027704  022225                  RXACK:  CMP     (R2)+,(R5)+             ;CHECK TX AND RX TABLE
027706  001015                          BNE     RXAERR
027710  005304                          DEC     R4
027712  001374                          BNE     RXACK

027714  062737  000400  012416          ADD     #400,LENGTH             ;CHANGE LENGTH
027722  022737  002000  012416          CMP     #2000,LENGTH
027730  100252                          BPL     TXABGN
027732  005303                          DEC     R3                      ;SELECT NEW PATERN
027734  001245                          BNE     TXATAR                  ;ALL DONE
027736  000137  030060                  JMP     RXAEND

027742  162705  000002          RXAERR: SUB     #2,R5
027746  162702  000002                  SUB     #2,R2

027752  011237  012410                  MOV     (R2),TXDATA
027756  011537  012412                  MOV     (R5),RXDATA

027762  005737  002250                  TST     FLAG                    ;LOOK IF 1ST ERROR
027766  00101'                          BNE     30$
```

```
4363 027770                                    ERRHRD  34,EM0017,PFRAME        ;DATA CMP ERROR
4364 030000  005237  002250                    INC     FLAG
4365 030004  062702  000002                    ADD     #2,R2                   ;POINT NEXT ADDRESS
4366 030010  062705  000002                    ADD     #2,R5
4367 030014  000137  027704                    JMP     RXACK
4368
4369 030020                            30$:    ERRHRD  34,0,PRAMEF             ;SHORT REPORT
4370 030030  005237  002250                    INC     FLAG
4371 030034  062702  000002                    ADD     #2,R2
4372 030040  062705  000002                    ADD     #2,R5                   ;POINT NEXT ADDRESS
4373 030044  022737  000010  002250            CMP     #10,FLAG                ;LOOK IF 10 REPORT
4374 030052  001314                            BNE     RXACK
4375
4376 030054                                    ESCAPE  TST
4377
4378
4379
4380
4381
4382 030060                            RXAEND:
4383 030060                            ENDTST
```

```
4385 030062                     BADHEAD
                                ;********************** TEST10 **************
4386                            ;TRANSMIT DIFFERENT FRAMES OF VARIOUS LENGTH IN EXTERNAL LOOP BACK
4387                            ;MODE ON CHANNEL B
4388 030062                     BADHEAD
                                ;********************** TEST10 **************
4389
4390
4391
4392
4393
4394
4395
4396 030062                     STARS 1
4397                            ;
4398                            ;AT BEGINNING OF TEST ,CHECK IF LOOP BACK CONNECTORS ARE INSTALLED
4399                            ;OR NOT:IF NOT INSTALLED = EXIT TEST AND GIVE ERROR MESSAGE
4400                            ;                          **********************************
4401                            ;
4402                            ;
4403                            ;
4404                            ;
4405                            ;
4406                            ;QBUS WRITTE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMV11 CSR'S
4407                            ;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED (56KB)
4408                            ;
4409                            ;
4410                            ;
4411                            ;DCT11 EXECUTE THE TRANSFER IN EXTERNAL MODE ON CHA  AND WRITTE BACK
4412                            ;IN RX TABLE
4413                            ;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
4414                            ;RX TABLE =TX TABLE
4415                            ;
4416                            ;
4417                            ;PARAMETERS SELECTION:
4418                            ;       SEL2=   TX TABLE ADDRESS
4419                            ;       SEL4=   TX TABLE LENGTH
4420                            ;       BSEL6=  EXTENDED ADDRESS OF TX TABLE
4421                            ;       BSEL7=      "       "       RX   "
4422                            ;       SEL12=  RX TABLE ADDRESS
4423                            ;       SEL14=  SPEED SELECTION
4424                            ;       SEL16=  ERROR STATUS
4425                            ;       BSEL0=  TEST STATUS
4426                            ;       SEL10=  RECEIVED BYTE COUNT     DIFFERENCE BETWEEN RX AND TX TABLE
4427                            ;                                       >0 IF TX>RX,<0 IF TX<RX
4428                            ;
4429                            ;
4430                            ;
4431                            ;
4432                            ;TEST STATUS DESCRIPTION:
4433                            ;       BSEL0=  0       =TEST DONE CHECK RX TABLE
4434                            ;       BSEL0=  200     =TIMEOUT ERROR
4435                            ;       BSEL0=  TSTNB   =NO KMV11 ANSWER
4436                            ;       BSEL0=  100     =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16
4437                            ;
4438                            ;
4439                            ;
```

```
4440                    ;ERROR STATUS DESCRIPTION:
4441                    ;
4442                    ;       WHEN BSEL0=100,GIVE CONTAINT OF ERROR STATUS AND WORD COUNT DISCREPANCY
4443                    ;
4444                    ;
4445                    ;       SEL16=  BIT14=1 =FCS ERROR
4446                    ;       SEL16=  BIT13=1 =OVERRUN ERROR
4447                    ;       SEL16=  BIT8 =1 =ILLEGAL INTERUPT ERROR
4448                    ;       SEL16=  BIT7 =1 =RX ABORT ERROR
4449                    ;       SEL16=  BIT6 =1 =UNDERRUN ERROR
4450                    ;       SEL16=  BIT5 =1 =WORD COUNT DISCREPANCY
4451                    ;       SEL16=  BIT4 =1 =DMA IN TIMEOUT ERROR
4452                    ;       SEL16=  BIT3 =1 =DMA OUT TIMEOUT ERROR
4453                    ;       SEL16=  BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
4454                    ;                                               ONLY DURING SELF TEST)
4455                    ;
4456                    ;
4457                    ;
4458                    ;MICRO DIAG TEST DESCRIPTION:
4459                    ;TEST 43        =TRANSMIT VARIOUS LENGTHFRAME AT 56 KBAUDS SPEED ON CHANNEL B
4460                    ;                       IN EXTERNAL LOOP BACK MODE
4461                    ;
4462                    ;
4463                    ;
4464                    ;
4465                    ;CAUTION:
4466                    ;--------
4467                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4468                    ;
4469                    ;
4470                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4471                    ;
4472                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4473                    ;EXTERNAL LOOP BACK CONECTOR
4474                    ;
4475                    ;
4476                    ;
4477                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4478                    ;MODEM CABLE.
4479                    ;
4480                    ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
4481                    ;MODEM CABLE.
4482                    ;CAUTION:
4483                    ;--------
4484                    ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
4485                    ;
4486                    ;
4487                    ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
4488                    ;
4489                    ;TO BE FULLY TESTED ,KMV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
4490                    ;EXTERNAL LOOP BACK CONECTOR
4491                    ;
4492                    ;
4493                    ;
4494                    ;FOR RS422 PLUG LOOP BACK CONNECTOR 2P-E155A-00 AT THE END OF 2P-E14UA-00
4495                    ;MODEM CABLE.
4496                    ;
```

```
                                        ;FOR RS423 PLUG LOOP BACK CONNECTOR  H325 AT THE END OF 2P-E14VA-00
                                        ;MODEM CABLE.
                                        ;
                                        ;
                                        ;
                                        ;
4503 030062                             STARS 1
```

```
4505 030062                            BGNTST
4506 030062  004737  014400                    JSR     PC,CLRKMV
4507 030066  004737  014600                    JSR     PC,CKKMV        ;LOOK IF LOOP BACK CON INSTALLED?
4508
4509
4510 030072  005737  012474                    TST     LOOP            ;IS LOOP BIT=1?
4511 030076  001412                            BEQ     BGNTXD          ;YES GO ON TEST
4512 030100                                    PRINTF  #MLOOP          ;NO LOOP BACK PLUGGED .THE MODULE
4513                                                                   ;WILL NOT BE TESTED IS EXTERNAL LOOP
4514
4515 030120  000137  030552                    JMP     RXDEND
4516
4517
4518
4519 030124  004737  014502          BGNTXD: JSR     PC,MAINM1       ;SET MAINT MODE
4520
4521 030130  012703  000005                    MOV     #5,R3           ;SELECT RANDOM PATTERN
4522 030134  012737  000174  012414            MOV     #KB56,TSPEED    ;SELECT SPEED
4523
4524 030142  012737  000001  012416  TxT AR:  MOV     #1,LENGTH       ;1ST TABLE LENGTH
4525
4526 030150  013704  012416          TXDBGN: MOV     LENGTH,R4
4527 030154  012702  002370                    MOV     #TTABLE,R2
4528 030160                                    BREAK
4529 030162  004737  013160          10$:    JSR     PC,GENER        ;WRITE TABLE
4530 030166  013722  012400                    MOV     DATA,(R2)+
4531 030172  005304                            DEC     R4
4532 030174  001372                            BNE     10$
4533
4534
4535 030176  013704  012416                    MOV     LENGTH,R4       ;CLEAR RX TABLE
4536 030202  012702  006370                    MOV     #RTABLE,R2
4537 030206  005022                  20$:      CLR     (R2)+
4538 030210  005304                            DEC     R4
4539 030212  001375                            BNE     20$
4540
4541
4542
4543
4544
4545 030214  013777  012414  162246          MOV     TSPEED,@KMVP14  ;SEND TX SPEED
4546 030222  012777  002370  162226          MOV     #TTABLE,@KMVP02 ;   "  TX TABLE ADDRESS
4547 030230  013777  012416  162222          MOV     LENGTH,@KMVP04  ;   "  "     "  LENGTH
4548 030236  012777  006370  162222          MOV     #RTABLE,@KMVP12 ;SEND RX TABLE ADDRESS
4549 030244  005077  162212                  CLR     @KMVP06         ;CLR EXTENDED ADDRESS
4550
4551
4552
4553
4554
4555 030250  004537  014552          1$:     JSR     R5,TSTNUB
4556 030254  000043                          .WORD   43              ;DO TEST 43= CHB TEST
4557
4558
4559
4560 030256                           2$:     WAITB   0,3             ;WAIT FOR TEST EXECUTION
4561
```

```
4562
4563 030276  004737  013102                JSR     PC,TSTERR               ;CHECK BSEL0
4564
4565 030302  000427                        BR      6$                      ;TEST OK CHECK RX TABLE
4566 030304  000402                        BR      3$                      ;TIMEOUT ERROR
4567 030306  000401                        BR      3$                      ;NO KMV11 ANSWER
4568 030310  000410                        BR      4$                      ;CHECK SEL16 TO SEE WHICH ONE
4569
4570
4571 030312                          3$:   ERRHRD  36,EM0004               ;NO KMV11 ANSWER
4572 030322  004737  012740                JSR     PC,CHKMAX               ;TOO MANY ERROR, DROP IF YES
4573 030326                                ESCAPE  TST
4574
4575
4576 030332                          4$:                                   ;ERROR DURING TEST READ ERROR STATUS
4577                                                                       ;TO CHECK WHICH ONE
4578
4579 030332  017737  162134  012424        MOV     @KMVP16,STAERR          ;READ ERROR STATUS
4580
4581 030340  017737  162120  012426        MOV     @KMVP10,WRDCNT          ;READ WORD COUNT DISCREPANCY
4582
4583 030346                                ERRHRD  37,EM0020,PRSTER        ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
4584                                                                       ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
4585 030356                                ESCAPE  TST
4586
4587
4588
4589
4590
4591 030362  012702  002370          6$:   MOV     #TTABLE,R2
4592 030366  012705  006370                MOV     #RTABLE,R5
4593 030372  013704  012416                MOV     LENGTH,R4
4594 030376  022225                  RXDCK: CMP     (R2)+,(R5)+
4595 030400  001015                        BNE     RXDERR
4596 030402  005304                        DEC     R4
4597 030404  001374                        BNE     RXDCK
4598
4599 030406  062737  000400  012416        ADD     #400,LENGTH
4600 030414  022737  002000  012416        CMP     #2000,LENGTH
4601 030422  100252                        BPL     TXDBGN
4602
4603 030424  005303                        DEC     R3
4604 030426  001245                        BNE     TXDTAR
4605 030430  000137  030552                JMP     RXDEND
4606
4607
4608
4609 030434  162705  000002          RXDERR: SUB   #2,R5
4610 030440  162702  000002                SUB     #2,R2
4611
4612 030444  011237  012410                MOV     (R2),TXDATA
4613 030450  011537  012412                MOV     (R5),RXDATA
4614
4615 030454  005737  002250                TST     FLAG                    ;LOOK IF 1ST ERROR
4616 030460  001014                        BNE     30$
4617
4618 030462                                ERRHRD  38,EM0015,PFRAME        ;DATA CMP ERROR
```

```
4619 030472  005237  002250                    INC     FLAG
4620 030476  062702  000002                    ADD     #2,R2                   ;POINT NEXT ADDRESS
4621 030502  062705  000002                    ADD     #2,R5
4622 030506  000137  030376                    JMP     RXDCK
4623
4624 030512                            30$:    ERRHRD  38,0,PRAMEF             ;SHORT REPORT
4625 030522  005237  002250                    INC     FLAG
4626 030526  062702  000002                    ADD     #2,R2
4627 030532  062705  000002                    ADD     #2,R5                   ;POINT NEXT ADDRESS
4628 030536  022737  000010  002250            CMP     #10,FLAG                ;LOOK IF 10 REPORT
4629 030544  001314                            BNE     RXDCK
4630
4631 030546                                    ESCAPE  TST
4632
4633
4634
4635
4636
4637 030552                            RXDEND:
4638 030552                            ENDTST
```

```
030554                         BADHEAD
                               ;*********************** TEST11 **************
                               ;TEST MODEM SIGNAL 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 WRITTE 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 BEINSTALLED
                               ;
                               ;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

```

```
                                BGNTST
030554  004737  014400                JSR     PC,CLRKMV               ;CLEAR KMV11 REGISTERS
030560  004737  014600                JSR     PC,CKKMV                ;LOOK IF KMV11A OR B AND IF LOOP BACK

030564  005737  012474                TST     LOOP                    ;LOOK IF LOOP BACK?
030570  001412                      BEQ     MOD108                  ;YES GO ON
030572                              PRINTF  #MLOOP                  ;NO LOOP BACK PLUGGED .THE MODULE
                                                                    ;WILL NOT BE TESTED IS EXTERNAL LOOP

030612  000137  031012              JMP     MODEND


030616  004737  014502        MOD108: JSR     PC,MAINM1

030622  012737  000100  030636        MOV     #100,MODWR1+6           ;WRITE TTL 108B


030630  004537  014702        MODWR1: JSR     R5,WRITE                ;WRITE KMV REG ADDRESS 130012
030634  130012                        .WORD   130012
030636  000000                        .WORD   0                       ;DATA TO WRITE

030640                              WAITA   0


030652  004537  014730                JSR     R5,READ                 ;READ KMV ADDRESS 130000
030656  130000                        .WORD   130000


030660  042737  177767  012374        BIC     #177767,BAD             ;MASK BIT BUT CCITT 107A/B
030666  022737  000010  012374        CMP     #10,BAD                 ;CMP 108B AND 107B
030674  001036                      BNE     MODERB                  ;REPORT ERROR IF BAD


030676  012737  000200  030712        MOV     #200,MODWR2+6           ;WRITE TTL 108A


030704  004537  014702        MODWR2: JSR     R5,WRITE                ;WRITE KMV REG ADDRESS 130012
030710  130012                        .WORD   130012
030712  000000                        .WORD   0                       ;DATA TO WRITE

030714                              WAITA   0


030726  004537  014730                JSR     R5,READ                 ;READ KMV ADDRESS 130000
030732  130000                        .WORD   130000


030734  042737  177737  012374        BIC     #177737,BAD             ;MASK BIT BUT CCITT 107A/B
030742  022737  000040  012374        CMP     #40,BAD                 ;CMP 108A AND 107A
030750  001420                      BEQ     MODEND                  ;OK EXIT TEST
```

```
                                
                                
                                
030752                          MODERA: ERRHRL  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 BSELO
                        ;IF ERROR SET 100 IN BSELO 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
                        ;
```

```
                                ;
                                ;CAUTION:
                                ;--------
                                ;RUN ONLY WITh EXTERNAL LOOP BACK CONNECTOR:
                                ;
                                ;
                                ;NOTE:FOR KMV11-B BOTH CONNECTORS MUST BEINSTALLED
                                ;
                                ;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.
                                ;
031014                          STARS 1
```

```
                                        BGNTST
                                                JSR     PC,CLRKMV       ;CLEAR ALL REGISTERS
                                                JSR     PC,CKKMV        ;TEST IF LOOP BACK CONNECTOR

                                                TST     LOOP
                                                BEQ     1$              ;LOOP BACK PRESENT GO ON

                                                PRINTF  #MLOOP          ;NO LOOP BACK PLUGGED .THE MODULE
                                                                        ;WILL NOT BE TESTED IS EXTERNAL LOOP

                                                JMP     MODEND          ;GO TO FOLLOWING TEST


                                        1$:
                                        BGNSUB

                                                JSR     PC,MAINM1       ;SET MAINTENANCE MODE
                                                JSR     R5,TSTNUB
                                                .WORD   34              ;SEND TEST 34(MODEM SIGNAL ON CHA)

                                                WAITB   0,2

                                                JSR     PC,TSTERR       ;CHECK TEST RESULT
                                                BR      3$              ;TEST OK GO ON
                                                BR      4$              ;TIMEOUT
                                                BR      4$              ;NO TEST ANSWER
                                                BR      5$              ;ERROR DURING TEST ,LOOK WHICH ONE


                                        4$:     ERRHRD  42,EM0004       ;NO ANSWER
                                                JSR     PC,CHKMAX       ;DROP IF TOO MANY ERROR
                                                ESCAPE  SUB

                                        5$:     MOV     @KMVP02,GOOD    ;READ WHICH SIGNAL WAS TESTED
                                                MOV     @KMVP04,BAD     ;   ''     ''    IS THE RESULT OF TEST
                                                MOV     @KMVP10,DATA    ;READ SIGAL VALUE

                                                ERRHRD  43,EM0032,PMODEM ;REPORT ERROR
                                                ESCAPE  SUB

                                        3$:
                                        ENDSUB


                                        BGNSUB

                                                JSR     PC,MAINM1       ;SET MAINTENANCE MODE
                                                JSR     R5,TSTNUB
                                                .WORD   35              ;SEND TEST 35(MODEM SIGNAL ON CHB)

                                                WAITB   0,2

                                                JSR     PC,TSTERR       ;CHECK TEST RESULT
                                                BR      3$              ;TEST OK GO ON
                                                BR      4$              ;TIMEOUT
                                                BR      4$              ;NO TEST ANSWER
```

```
4898 031254  000410                          BR      5$                      ;ERROR DURING TEST ,LOOK WHICH ONE
4899
4900
4901
4902 031256                            4$:    ERRHRD  44,EM0004               ;NO ANSWER
4903 031266  004737  012740                   JSR     PC,CHKMAX               ;DROP IF TOO MANY ERROR
4904 031272                                   ESCAPE  SUB
4905
4906 031276  017737  161154  002272  5$:    MOV     @KMVP02,GOOD            ;READ WHICH SIGNAL WAS TESTED
4907 031304  017737  161150  012374          MOV     @KMVP04,BAD             ;   "     "    IS THE RESULT OF TEST
4908 031312  017737  161146  012400          MOV     @KMVP10,DATA            ;READ SIGAL VALUE
4909
4910 031320                                   ERRHRD  45,EM0034,PMODEM        ;REPORT ERROR
4911 031330                                   ESCAPE  SUB
4912
4913 031334                            3$:
4914 031334                            ENDSUB
4915
4916
4917 031336                            ENDTST
```

```
                                .SBTTL HARDWARE PARAMETER CODING SECTION



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

031340                                  BGNHRD

031342                                  GPRMA   ADDRES,0,0,60000,177776,YES
031352                                  GPRMA   VECTOR,2,0,0,674,YES
031362                                  GPRMD   PRIRTY,4,0,7000,4,7,YES
031374                                  ENDHPD


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
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
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
```

```
4956
4957                                    .EVEN
4958
4959
4960
4961
4962
4963
```

```
                                .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
```

```
   5001
   5002 031520                          $PATCH::
   5003 031520                                  .BLKW   50
   5004
   5011
   5012 031640                                  LASTAD
        031644                          L$LAST::
   5013 031644                                  ENDMOD
   5014
   5015
```

```
5017
5018
5031
5032 031644                                    BGNSETUP        1
5033 031644                                    BGNPTAB
5034 031650  177000                            .WORD   177000
5035 031652  000300                            .WORD   300
5036 031654  004000                            .WORD   4000
5037 031656  000001                            .WORD   1
5038 031660                                    ENDPTAB
5039 031660                                    ENDSETUP
5040
5041
5042
5043
5044
5045         000001                .END
```

SYMBOL TABLE

```
ABORT   023522          CKSELO  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$CVEC= 000036          EM0002  015214          F$SUB = 000002          INTFLG  012372
BIT01 = 000002 G        C$DCLN= 000044          EM0003  015257          F$SW  = 000014          INTTX   025244
BIT02 = 000004 G        C$DODU= 000051          EM0004  015345          F$TEST= 000001          ISR   = 000100 G
BIT03 = 000010 G        C$DRPT= 000024          EM0006  015374          GDDAT   002364          IXE   = 004000 G
BIT04 = 000020 G        C$DU  = 000053          EM0007  015436          GDPEV   012430          I$AU  = 000041
BIT05 = 000040 G        C$EDIT= 000003          EM0010  015533          GENER   013160          I$AUTO= 000041
BIT06 = 000100 G        C$ERDF= 000055          EM0011  015625          GENER1  013302          I$CLN = 000041
BIT07 = 000200 G        C$ERHR= 000056          EM0012  015704          GENEX   013440          I$DU  = 000041
BIT08 = 000400 G        C$ERRO= 000060          EM0013  016002          GENINC  013432          I$HRD = 000041
BIT09 = 001000 G        C$ERSF= 000054          EM0014  016312          GENISH  013310          I$INIT= 000041
BIT1  = 000002 G        C$ERSO= 000057          EM0015  016363          GENO    025170          I$MOD = 000041
BIT10 = 002000 G        C$ESCA= 000010          EM0016  016466          GENOUT  024722          I$MSG = 000041
BIT11 = 004000 G        C$ESEG= 000005          EM0017  016751          GENRAN  013312          I$PROT= 000040
BIT12 = 010000 G        C$ESUB= 000003          EM0020  017046          GENROT  013266          I$PTAB= 000041
BIT13 = 020000 G        C$ETST= 000001          EM0023  017143          GENR0   013254          I$PWR = 000041
BIT14 = 040000 G        C$EXIT= 000032          EM0024  017217          GENR1   013244          I$RPT = 000041
BIT15 = 100000 G        C$GETB= 000026          EM0027  017246          GENSEL  013176          I$SEG = 000041
BIT2  = 000004 G        C$GETW= 000027          EM0030  017320          GEN0    013216          I$SETU= 000041
BIT3  = 000010 G        C$GMAN= 000043          EM0032  017512          GEN1    013222          I$SFT = 000041
BIT4  = 000020 G        C$GPHR= 000042          EM0033  016242          GEN25   013236          I$SRV = 000041
BIT5  = 000040 G        C$GPLO= 000030          EM0034  017603          GEN52   013230          I$SUB = 000041
BIT6  = 000100 G        C$GPRI= 000040          EM0035  017674          GETPRM  023320          I$TST = 000041
BIT7  = 000200 G        C$INIT= 000011          EM0112  016063          GOOD    002272          J$JMP = 000167
BIT8  = 000400 G        C$INLP= 000020          EM0113  016161          GOOD0   002274          KB1.2 = 013224 G
BIT9  = 001000 G        C$MANI= 000050          EM0115  016556          GOOD1   002276          KB2.4 = 005473 G
BOE   = 000400 G        C$MEM = 000031          EM0116  016654          GOOD10  002306          KB56  = 000174 G
BRXCK   026122          C$MSG = 000023          EM0130  017415          GOOD12  002310          KB64  = 000154 G
BRXEND  026262          C$OPEN= 000034          END     023530          GOOD14  002312          KB68  = 000146 G
BRXERR  026144          C$PNTB= 000014          ERRBLK  002230 G        GOOD16  002314          KB70  = 000143 G
BSEL0   012376          C$PNTF= 000017          ERRCNT  002236          GOOD2   002300          KB72  = 000141 G
BSEL1   002340          C$PNTS= 000016          ERRMSG  002226 G        GOOD4   002302          KIND    012404
BTXSTA  025674          C$PNTX= 000015          ERRNBR  002224 G        GOOD6   002304          KMTLVL  012452
BXCK  = ****** GX       C$QIO = 000377          ERRTYP  002222 G        G$CNTO= 000200          KMVCSR  012454
BXLBGN  026760          C$RDBU= 000007          EVL   = 000004 G        G$DELM= 000372          KMVLVL  012442
BXLCK   027206          C$REFG= 000047          EXADDR  012370          G$DISP= 000003          KMVP02  012456
BXLEND  027362          C$RESE= 000033          E$END = 002100          G$EXCP= 000400          KMVP04  012460
BXLERR  027244          C$REVI= 000003          E$LOAD= 000035          G$HILI= 000002          KMVP06  012462
BXLTAR  026752          C$RFLA= 000021          FLAG    002250          G$LOLI= 000001          KMVP10  012464
CBSEL0  013652          C$RPT = 000025          FTIME   002252          G$NO  = 000000          KMVP12  012466
CHANEL  012406          C$SEFG= 000046          F$AU  = 000015          G$OFFS= 000400          KMVP14  012470
CHKMAX  012740          C$SPRI= 000041          F$AUTO= 000020          G$OFSI= 000376          KMVP16  012472
CKALL   013710          C$SVEC= 000037          F$BGN = 000040          G$PRMA= 000001          KMVV00  012440
CKKMV   014600          C$TPRI= 000013          F$CLEA= 000007          G$PRMD= 000002          KMVV02  012446
CKREG   014156          DATA    012400          F$DU  = 000016          G$PRML= 000000          KMVV04  012444
```

SYMBOL TABLE

```
KMVV06  012450          L$SPTP  002024 G        MINTR   020620          PRREG   022120 G        SSTACK  012676
KMV11B  002000 G        L$STA   002030 G        MLOOP   017736          PRSELO  021516 G        STAERR  012424
LENGTH  012416          L$SW    002260          MODEM1  021300          PRSTER  023004 G        SVCGBL= 000000
LOCK    002232          L$TEST  002114 G        MODEM2  021344          PRT11V  022524 G        SVCINS= 177777
LOE   = 040000 G        L$TIML  002014 G        MODEM3  021403          PSTACK  002242          SVCSUB= 177777
LUGDEV  002240          L$UIT   002262          MODEND  031012          PVECT   022466 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  021552          MODWR1  030630          RANDN   002350          TFM36   015032
L$ACP   002110 G        L10003  021610          MODWR2  030704          RANEX   013430          TIM     015014
L$APT   002036 G        L10004  022116          MOD108  030616          RANGEN  013332          TSPEED  012414
L$AU    023772 G        L10005  022374          MRAMEF  021446          RANMTA  002346          TSTERR  013102
L$AUT   002070 G        L10006  022432          MREG0   020062          RANSEC  013416          TSTNUB  014552
L$AUTO  023614 G        L10007  022464          MREG10  020302          RANSEL  002344          TTABLE  002370
L$CCP   002106 G        L10010  022522          MREG12  020346          RANST   002342          TXABGN  027456
L$CLEA  023706 G        L10011  022560          MREG14  020412          RAN1    013344          TXATAR  027450
L$CO    002032 G        L10012  022646          MREG16  020456          RAN2    013362          TXDATA  012410
L$DEPO  002011 G        L10013  022750          MREG2   020126          RAN4    013420          TXDBGN  030150
L$DESC  002174 G        L10014  023002          MREG4   020172          READ    014730          TXDTAR  030142
L$DESP  002076 G        L10015  023060          MREG6   020236          REGADR  012476          TXLBGN  026320
L$DEVP  002060 G        L10016  023116          MSEL0   020014          REVPRO  024126          TXLTAR  026312
L$DISP  002132 G        L10017  023124          MSTER1  021203          ROMMAP  023774          TXSTAR  025250
L$DLY   002116 G        L10020  023530          MSTER2  021235          RSTREG  013540          T$ARGC= 000001
L$DTP   002040 G        L10021  023704          MT11V   020760          RTABLE  006370          T$CODE= 002032
L$DTYP  002034 G        L10022  023710          MVECT   020704          RTCLK   024176          T$ERRN= 000055
L$DU    023712 G        L10023  023770          NERRS   013032          RUNNIN  023532          T$EXCP= 000000
L$DUT   002072 G        L10024  023772          NEXT    023240          RXACK   027704          T$FLAG= 000040
L$DVTY  012676 G        L10025  024114          NUB     012420          RXAEND  030060          T$FREE= 031660
L$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$HPTP  002022 G        L10040  030552          PADFLT  023062 G        RXLEND  026722          T$PCNT= 000000
L$HW    002164 G        L10041  031012          PBSEL0  022376 G        RXLERR  026604          T$PTAB= 010050
L$ICP   002104 G        L10042  031336          PFRAME  022562 G        SAVE4   002254          T$PTHV= 000001
L$INIT  023126 G        L10043  031204          PINTR   022434 G        SAVE6   002256          T$PTNU= 000001
L$LADP  002026 G        L10044  031334          PMODEM  022650 G        SAVPC   002246          T$SAVL= 177777
L$LAST  031644 G        L10045  031374          PNT   = 001000 G        SAVPC1  002352          T$SEGL= 177777
L$LOAD  002100 G        L10046  031520          PRALL   021612 G        SAVREG  013460          T$SIZE= 000006
L$LUN   002074 G        L10047  031650          PRAMEF  022752 G        SAVSP   002244          T$SUBN= 000002
L$MREV  002050 G        L10051  031660          PRI   = 002000 G        SAVSTA  002354          T$TAGL= 177777
L$NAME  002000 G        MAINM1  014502          PRINT   021554 G        SEL0    002316          T$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$$AU = 010024
L$RPT   023120 G        MCLR  = 040000 G        PRI04 = 000200 G        SEL2    002322          T$$AUT= 010021
L$SOFT  031520 G        MFRAM1  021042          PRI05 = 000240 G        SEL4    002324          T$$CLE= 010022
L$SPC   002056 G        MFRAM2  021120          PRI06 = 000300 G        SEL6    002326          T$$DAT= 010051
L$SPCP  002020 G        MINT    020522          PRI07 = 000340 G        SETUP   023232          T$$DU = 010023
```

```
KMV11 B LINE CNT  DIAG. MACRO M1200  06-JAN-83 09:42  PAGE 83-3
SYMBOL TABLE

T$$HAR= 010045          T$$SUB= 010044          T3      024166 G        UAM   = 000200 G        WRITE   014702
T$$HW = 010001          T$$TES= 010042          T4      024476 G        UNIT    002264           X$ALWA= 000000
T$$INI= 010020          T1      023774 G        T4.1    024476          UUT     012436           X$FALS= 000040
T$$MSG= 010016          T10     030062 G        T4.2    024744          VECT    012402           X$OFFS= 000400
T$$PC = 000001          T11     030554 G        T5      025214 G        VECTOR  031426           X$TRUE= 000020
T$$PRO= 010000          T12     031014 G        T6      025640 G        WAIT1   012726           $LSTIN= 000000
T$$PTA= 010050          T12.1   031056          T7      026264 G        WAIT2   012706           $LSTTA= 000000
T$$RPT= 010017          T12.2   031206          T8      026724 G        WRDCNT  012426           $PATCH  031520 G
T$$SOF= 010046          T2      024116 G        T9      027364 G

. ABS.  031650      000
        C00000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29096 WORDS  ( 114 PAGES)
DYNAMIC MEMORY:  21924 WORDS  ( 84 PAGES)
ELAPSED TIME:  00:26:08
VKMEA0.BIN,VKMEA0=[64,3]LIBA.MLB/ML,[64,5]VKMEA0
```